AFCRL-64-55

SCIENTIFIC REPORT NO. 3
CONTRACT NO. AF19(628)-2379
PROJECT NO. 7600
TASK NO. 760003

Differential Orbit Correction And Station Locator Program "DOC-II"

By

Richard M. Moroney

Isabel M. Hussey

FEBRUARY 1964

Prepared for

AIR FORCE CAMBRIDGE RESEARCH LABORATORIES
OFFICE OF AEROSPACE RESEARCH
UNITED STATES AIR FORCE
BEDFORD, MASSACHUSETTS

Wolf
Research and Development Corporation
P.O. Box 36; Baker Avenue, West Concord, Massachusetts

AFCRL-64-55

SCIENTIFIC REPORT NO. 3
CONTRACT NO. AF19(628)-2379
PROJECT NO. 7600
TASK NO. 760003

Differential Orbit Correction And Station Locator Program "DOC-II"

By

Richard M. Moroney

Isabel M. Hussey

FEBRUARY 1964

Prepared for

AIR FORCE CAMBRIDGE RESEARCH LABORATORIES
OFFICE OF AEROSPACE RESEARCH
UNITED STATES AIR FORCE
BEDFORD, MASSACHUSETTS

Wolf
Research and Development Corporation
P.O. Box 36; Baker Avenue, West Concord, Massachusetts

Requests for additional copies by Agencies of the Department of Defense, their contractors, and other Government agencies should be directed to the:

DEFENSE DOCUMENTATION CENTER (DDC)
CAMERON STATION
ALEXANDRIA, VIRGINIA

Department of Defense contractors must be established for DDC services or have their 'need-to-know' certified by the cognizant military agency of their project or contract.

All other persons and organizations should apply to the:

U.S. DEPARTMENT OF COMMERCE
OFFICE OF TECHNICAL SERVICES
WASHINGTON 25, D.C.

ABSTRACT

The analysis and programming associated with performing a differential correction to orbital elements and/or station positions from weighted observations of range, range rate, azimuth-elevation and/or right ascension-declination on close earth satellites are described. The ephemeris computation considers perturbations due to the earth's gravitational potential as described by a spherical harmonic representation through n = m = 5, the lunar gravitational potential, air drag and radiation pressure. All observations may be simulated in punched card form for the purposes of conducting geometric studies.

This contract was administered under the direction of the project scientists, Mr. A. Mancini and Captain L.L. Sheldon, of the Geodesy and Gravity Branch, Terrestrial Sciences Laboratory.

CONTRIBUTORS

Michael W. Bate
Harry R. Kahler
William T. Nixon
Gerald A. Ouellette
Richard T. Tear
John T. Tillson

TABLE OF CONTENTS

I INTRODUCTION

The multi-purpose program described herein, "DOC II," has been developed primarily to obtain geocentric positions of ground-based instrumentation sites from observations made on close earth satellites. The observations may be in the form of range, range rate, and/or directions (right ascension and declination, azimuth and elevation). Provision for weighting observations is included.

This program utilizes the analysis and much of the programming associated with the ephemeris computation and differential correction theory that is described in AFCRL 62-892, Research in Geodesy and Gravity, Computer Programs for Orbit Correction and Station Location. For ready reference this portion of 62-892 is contained in Appendix A. This original ephemeris computation has been modified to consider the following additional perturbations.

1. the sectorial and tesseral terms in the earth's gravitational potential (thru n = m = 5)
2. the lunar gravitational potential
3. solar radiation pressure

The station locator portion of 62-892 was completely reworked and is described in the body of this report. Further modifications made to the original program now allows one to:

1. Correct the orbital elements and station positions simultaneously.
2. Simulate observations in the form of punched card output. Using this as input allows one to observe the effect that varying amounts, distribution, accuracy, and types of data has on the end results.
3. Correct the six orbital elements only, applying from one to nine corrections.
4. Correct the station positions only for a number of stations contained in up to six different datums.
5. Generate orbital elements for each time step specified by a Δt from input.

II FORMULATION

A. General

1. Standard Keplerian elements are obtained by inverting:

$$L_o = M_o + \omega + \Omega$$

$$A_{xn} = e \cos \omega$$

$$A_{yn} = e \sin \omega$$

$$H_x = \sqrt{a(1-e^2)} \quad \sin \Omega \ \sin i$$

$$H_y = -\sqrt{a(1-e^2)} \quad \cos \Omega \ \sin i$$

$$H_z = \sqrt{a(1-e^2)} \quad \cos i$$

2. Greenwich sidereal time in degrees for January 0.0 is selected internally from a table for the years 1958 to 1977. The present table for θ_g is:

(1960) = 98.67401
(1961) = 99.420937
(1962) = 99.1822167
(1963) = 98.943500314

Greenwich sidereal time at the time of an observation is obtained by applying a mean rotation to GST of January 0.0 and then adding on the equation of equinoxes from a table. In the original ephemeris computation (Appendix A), the equation of equinoxes was not added.

3. The maximum size of the matrix is 24 x 24. This allows for correcting the 6 orbital elements and up to 6 different non-zero datums.

4. Observation sigmas are selected internally if none are given from input.
The program will use the following nominal values:

10 meters for Range Observations
10.0 cps for Range Rate Observations
.00001 radians for Optical Observations

B. Observation Weighting and Calculation of Standard Deviation (σ) in Satellite Position

1 Preliminaries

As noted in the writeup of the Station Corrector Portion (pg 6), each observation leads to a condition equation of the form

$$Q_{obs} - Q_{comp} = C_{\Delta i} \Delta i + \ldots\ldots + C_{\Delta Z_j} \Delta Z_j \tag{1}$$

Here Q just denotes some quantity. The actual quantity used for Q varies with the type of observation as follows:

Observation Type	Q
Range rate	$(\underline{\rho} \cdot \dot{\underline{\rho}})$
Range	$\sqrt{\underline{\rho} \cdot \underline{\rho}}$
Azimuth or Right Ascension	$(\underline{\rho} \cdot \underline{A})$
Elevation or Declination	$(\underline{\rho} \cdot \underline{D})$

The basis of the entire differential correction method is the assumption that

$$Q_{obs} = Q_{true} + \epsilon \tag{2}$$

where ϵ is a sample from a distribution with zero mean and known standard deviation. Since some observations are more accurate than others, i.e., have smaller σ's, the condition equations are weighted by dividing by a σ for the error in that observation.

An additional advantage of this weighting, aside from the obvious one of being able to take account of variable observational accuracy, is that the diagonal elements of the least-squares matrix can be interpreted as variances in the computed corrections Δi, $\Delta\Omega, \ldots\ldots, \Delta X$, ΔY, ΔZ. An additional assumption is necessary here, namely that the errors in different observations are independent.

The estimates of the variances in the elements at epoch can be converted into estimates of the variances in the computed x, y, z coordinates of the satellite.

2 Theory

a) Variances for x, y, z coordinates of satellite.

Formulas for converting Δa_0, Δi_0, ... , $\Delta\Omega_0$ to $\Delta x(t_i)$, $\Delta y(t_i)$, $\Delta z(t_i)$ for each observation time t_i are given in Reference 2. A standard technique then gives the variances for the conversion Matrix A.

$$\begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} = \begin{bmatrix} & & \\ & A & \\ & & \end{bmatrix} \begin{bmatrix} \Delta a_0 \\ \Delta i_0 \\ \vdots \\ \Delta\Omega_0 \end{bmatrix}$$

A covariance Matrix C for $[\Delta x, \Delta y, \Delta z]$ is then related to the covariance Matrix for $[\Delta a_0, \Delta i_0, \ldots\ldots, \Delta\Omega_0]$ by

$$C = ADA^T$$

b) Observational Sigmas

In order to get a σ for the ϵ in (2) one must take account of the nature of the observed quantities, Q. It is easy to see that

$$\begin{aligned} (\underline{\rho} \cdot \dot{\underline{\rho}}) &= r\dot{r} \\ \sqrt{\underline{\rho} \cdot \underline{\rho}} &= r, \\ \Delta(\underline{\rho} \cdot \underline{A}) &= \rho \cos\delta \ \Delta\alpha, \\ \Delta(\underline{\rho} \cdot \underline{D}) &= \rho\,\Delta\delta, \end{aligned}$$

Thus appropriate sigmas are

$$\begin{aligned} (\underline{\rho} \cdot \dot{\underline{\rho}}) &: r\sigma_{\dot{r}}, \\ \sqrt{\underline{\rho} \cdot \underline{\rho}} &: \sigma_r, \\ \Delta(\underline{\rho} \cdot \underline{A}) &: \rho\cos\delta\ \sigma_\alpha, \\ \Delta(\underline{\rho} \cdot \underline{D}) &: \rho\sigma_\delta. \end{aligned}$$

3 Additional Features

a) Nominal values of the sigmas for each station's observations of range rate, range and angles are accepted from the station data cards. If the appropriate columns of a station's card are zero the program uses, for that station, a set of arbitrarily chosen values. When an observation card is read the program checks to see whether a sigma for the observation is on the card. If there is such a sigma it is the one used. If the appropriate columns are zero the nominal sigma for the observing station is used.

b) The subroutine which forms the condition equations divides them by the appropriate σ as determined in the theory section.

c) At each observation time the program saves on tape those values necessary for computing the matrix A mentioned in the theory section.

d) At the conclusion of a run the tape record mentioned in (c) is read, and sigmas for x, y and z at each observation time computed.

e) In order to be assured of getting sigmas in x, y, and z at epoch — even when this is not an observation time — a record, as in paragraph (c) is written at the start of the run, i. e., at epoch.

f) Simulated data is available in punched card form. Format is that required for input to the simulated run with toggles set as noted in the section on input.

C. Station - Corrector Portion

1 Preliminaries

Let x, y, z be the inertial coordinates of the satellite at some sidereal time θ and X, Y, Z be the earth-fixed coordinates of an observing station. The inertial coordinates of this station, $\hat{X}$, $\hat{Y}$, $\hat{Z}$, are given by

$$\begin{aligned} \hat{X} &= X \cos\theta - Y \sin\theta \\ \hat{Y} &= X \sin\theta + Y \cos\theta \\ \hat{Z} &= Z \end{aligned} \tag{1}$$

The station-satellite vector, $\underline{\rho}$ is

$$\underline{\rho} = (x - \hat{X})\,\underline{i} + (y - \hat{Y})\,\underline{j} + (z - \hat{Z})\,\underline{k} \tag{2}$$

The time derivative of this vector, $\underline{\dot{\rho}}$, is

$$\underline{\dot{\rho}} = (\dot{x} - \dot{\hat{X}})\,\underline{i} + (\dot{y} - \dot{\hat{Y}})\,\underline{j} + (\dot{z} - \dot{\hat{Z}})\,\underline{k} \tag{3}$$

If we let the earth's rotation rate be ω and differentiate (1) we get

$$\begin{aligned} \dot{\hat{X}} &= (-X \sin\theta - Y \cos\theta)\,\omega \\ \dot{\hat{Y}} &= (X \cos\theta - Y \sin\theta)\,\omega \\ \dot{\hat{Z}} &= 0 \end{aligned} \tag{4}$$

Using (1) and (4) to get rid of the hat variables in (2) and (3) gives

$$\begin{aligned} \underline{\rho} = {} & (x - X\cos\theta + Y \sin\theta)\,\underline{i} + (y - X \sin\theta - Y\cos\theta)\,\underline{j} \\ & + (z - Z)\,\underline{k} \end{aligned} \tag{5}$$

$$\begin{aligned} \underline{\dot{\rho}} = {} & (\dot{x} + [\,X \sin\theta + Y\cos\theta\,]\,\omega\,)\,\underline{i} \\ & + (\dot{y} + [\,Y \sin\theta - X\cos\theta\,]\,\omega\,)\,\underline{j} \\ & + \dot{z}\,\underline{k} \end{aligned} \tag{6}$$

Thus

$$\frac{\partial \underline{\rho}}{\partial X} = -\cos\theta\,\underline{i} - \sin\theta\,\underline{j}$$

$$\frac{\partial \underline{\rho}}{\partial Y} = \sin\theta\,\underline{i} - \cos\theta\,\underline{j}$$

$$\frac{\partial \underline{\rho}}{\partial Z} = -\underline{k}$$

Also

$$\begin{aligned}(\underline{\rho}\cdot\underline{\dot{\rho}}) &= (x - X\cos\theta + Y\sin\theta)(\dot{x} + [X\sin\theta + Y\cos\theta]\,\omega) \\ &\quad + (y - X\sin\theta - Y\cos\theta)(\dot{y} + [Y\sin\theta - X\cos\theta]\,\omega) \\ &\quad + (z - Z)\,\dot{z} \\ &= X\,(-\cos\theta\,\dot{x} + \omega\sin\theta\,x - \sin\theta\,\dot{y} - \omega\cos\theta\,y) \\ &\quad + Y\,(\sin\theta\,\dot{x} + \omega\cos\theta\,x - \cos\theta\,\dot{y} + \omega\sin\theta\,y) \\ &\quad + Z\,(-\dot{z}) \\ &\quad + \text{terms independent of } X,\ Y,\ Z.\end{aligned}$$

Notice that

$$(\underline{\rho}\cdot\underline{\dot{\rho}}) = \tfrac{1}{2}\cdot\frac{d}{dt}(\underline{\rho}\cdot\underline{\rho}) = \tfrac{1}{2}\frac{d}{dt}(\text{range})^2 = (\text{range})(\text{range rate}).$$

Here

$$\frac{\partial(\underline{\rho}\cdot\underline{\dot{\rho}})}{\partial X} = -\cos\theta\,(\dot{x} + \omega y) + \sin\theta\,(\omega x - \dot{y})$$

$$\frac{\partial(\underline{\rho}\cdot\underline{\dot{\rho}})}{\partial Y} = \cos\theta\,(\omega x - \dot{y}) + \sin\theta\,(\dot{x} + \omega y)$$

$$\frac{\partial(\underline{\rho}\cdot\underline{\dot{\rho}})}{\partial Z} = -\dot{z}$$

2 Theory

For each type of observation the original program (Appendix A) developed a condition equation. These were of the following forms:

Range Rate Observation

$$(\underline{\rho}\cdot\dot{\underline{\rho}})_{obs} - (\underline{\rho}\cdot\dot{\underline{\rho}})_{comp} = \frac{\partial(\underline{\rho}\cdot\dot{\underline{\rho}})}{\partial i}\,\Delta i + \ldots . + \frac{\partial(\underline{\rho}\cdot\dot{\underline{\rho}})}{\partial\Omega}\,\Delta\Omega$$

Azimuth or Right Ascension Observation

$$(\underline{\rho}\cdot\underline{A})_{obs} - (\underline{\rho}\cdot\underline{A})_{comp} = \frac{\partial(\underline{\rho}\cdot\underline{A})}{\partial i}\,\Delta i + \ldots + \frac{\partial(\underline{\rho}\cdot\underline{A})}{\partial\Omega}\,\Delta\Omega$$

Elevation or Declination Observation

$$(\underline{\rho}\cdot\underline{D})_{obs} - (\underline{\rho}\cdot\underline{D})_{comp} = \frac{\partial(\underline{\rho}\cdot\underline{D})}{\partial i}\,\Delta i + \ldots . + \frac{\partial(\underline{\rho}\cdot\underline{D})}{\partial\Omega}\,\Delta\Omega$$

Range Observation

$$\sqrt{(\underline{\rho}\cdot\underline{\rho})}_{obs} - \sqrt{(\underline{\rho}\cdot\underline{\rho})}_{comp} = \frac{\partial\sqrt{(\underline{\rho}\cdot\underline{\rho})}}{\partial i}\,\Delta i + \ldots . + \frac{\partial\sqrt{(\underline{\rho}\cdot\underline{\rho})}}{\partial\Omega}\,\Delta\Omega$$

Here the subscripts "obs" and "comp" denote "observed" and "computed" respectively. $\underline{A}$ and $\underline{D}$ are unit vectors in a plane perpendicular to ρ_{obs} and perpendicular to each other (Appendix A).

To modify these condition equations when X, Y and Z are also unknown, we must add to the right side the partials of the relevant quantity with respect to X, Y, and Z. We have already computed $\partial(\underline{\rho}\cdot\dot{\underline{\rho}})$ with respect to X, Y and Z. In the case of $(\underline{\rho}\cdot\underline{D})$ or $(\underline{\rho}\cdot\underline{A})$

$$\frac{\partial(\underline{\rho}\cdot\underline{D})}{\partial X} = \left(\frac{\partial\underline{\rho}}{\partial X}\cdot\underline{D}\right)$$

$$\frac{\partial(\underline{\rho}\cdot\underline{A})}{\partial Y} = \left(\frac{\partial\underline{\rho}}{\partial Y}\cdot\underline{A}\right)$$

etc. and the partials of $\underline{\rho}$ are known. Finally,

$$\frac{\partial\sqrt{\underline{\rho}\cdot\underline{\rho}}}{\partial X} = \frac{1}{2\sqrt{\underline{\rho}\cdot\underline{\rho}}}\;\frac{\partial(\underline{\rho}\cdot\underline{\rho})}{\partial X}$$

$$= \frac{1}{2\sqrt{\underline{\rho}\cdot\underline{\rho}}}\;2\left(\frac{\partial\underline{\rho}}{\partial X}\cdot\underline{\rho}\right)$$

$$= \left(\frac{\partial \underline{\rho}}{\partial X} \cdot \underline{L}\right)$$

where $\underline{L} = \underline{\rho} / \sqrt{\underline{\rho} \cdot \underline{\rho}}$ is a unit vector from station to satellite.

$X \rightarrow Y, Z.$

3. Modifications To AFCRL 62-892

a) The first change in the program causes it to accept a datum number on each station card. Up to six different non-zero datum numbers may appear in the deck of station cards for any run. (These need not be numbered in order such as 1, 2, 3. The program reassigns internal datum numbers so that the input datum numbers could be 12, 24, 268 etc.) A zero datum number indicates that a station is to be held fixed. All stations in the same datum are moved together.

The program counts the number of non-zero datums present, say n, and sets the number of unknowns equal to 6 + 3n.

b) The second change is a revision of the routine which forms the condition equations. These are enlarged as described in the theory section. An essential part of this step is to determine the datum number of the observing station, so as to make the partials with respect to X, Y and Z refer to X, Y and Z of the relevant datum.

To make clear what is meant here, suppose that there are two non-zero datums numbered 1 and 2. There are twelve unknowns,

$$\frac{\Delta a}{a}, \Delta i, \ldots, \Delta\Omega, \Delta X_1, \Delta Y_1, \Delta Z_1, \Delta X_2, \Delta Y_2, \Delta Z_2.$$

Each individual observation is made by a station in datum zero or one or two. If the station is in datum zero the coefficients of ΔX_1, $\Delta Y_1, \ldots, \Delta Z_2$ are all zero. If the station is in datum one the coefficients of ΔX_2, ΔY_2 and ΔZ_2 are zero but the coefficients of ΔX_1, ΔY_1, ΔZ_1 are those we just derived.

c) A third change in the program consists of a bookkeeping chore. After a set of datum shifts is determined, the stations in non-zero datums must have their X, Y, Z coordinates updated. This involves rewriting the observation tape.

III PROCESSING

A. The general flow of the program during execution is:

1. Set up constants; read in the first five specification cards.

2. Compute, for epoch, the values of inclination, node, W Bar, mean motion, A Bar.

3. Set up to integrate ephemeris.

4. Write parameters used in computing standard deviations in satellite position at epoch on Tape 11.

5. Integrate ephemeris:

   a. Compute time derivatives of A Bar, H Bar, and XL.

   b. Compute perturbative accelerations and relate to orbital elements.

   c. Compute required derivatives.

   d. After each perigee passage, correct orbit parameters for atmospheric drag.

   e. Transfer current values of A Bar, H Bar, and XL to ephemeris buffer.

   f. Determine what to print for output.

   g. Write ephemeris buffer on Tape 3 whenever it is filled.

6. Determine if Simulation Run or Differential Correction Run.

If Simulation Run

Compute and print simulated observations for each station - Range, Range Rate, Right Ascension/Declination, Azimuth/Elevation.

If Differential Correction Run

1. Read station cards.

2. Set up least squares routine for matrix size.

3. For each observation card perform the following:

a. Read an observation card.

b. Match station number on observation card with station card.

c. Check if observation time is within ephemeris time, i. e., between T_o and T_f.

d. Compute L Bar, etc.

e. Save observation data on Tape 9.

4. When all observation cards have been read in, continue processing each observation:

   a. Find observation time in ephemeris.

   b. Interpolate to get parameters from ephemeris.

   c. Compute x, y, z, and related parameters.

   d. Save x, y, z, $\dot{x}$, $\dot{y}$, $\dot{z}$ on Tape 6.

   e. Write parameters for standard deviation for each observation time on Tape 11.

   f. Compute correction equation coefficients and residuals.

   g. Update sum of squares of residuals and normal equations.

5. Save matrix A and B as determined by input code to correct both elements and station positions, elements only, or station positions only.

6. Solve normal equations.

7. Compute root mean square of residuals; reset limit for acceptance of residuals.

8. Determine whether or not to do another pass.

9. Apply correction to elements.

10. Apply correction to station position.

11. Determine whether or not to do another correction.

12. Compute the sigmas in x, y, and z (position of observer).

13 Output

14. Exit

B. Error Indicators

1. Subroutine Error

This is a general error exit from any subroutine and causes the execution of the program to stop. Location SUBER 1 contains the address within the subroutine from which the exit occurred.

2. Altitude Below 50000 Meters

Program exits. The internal table of altitudes has a lower limit of 50000 meters and an upper limit of 900000 meters.

3. Kepler's Equation Not Converging For "e"

Program exits. This error occurs as the value of eccentricity approaches 1.0.

4. Error in Range Kutta Routine

Program exits.

5. Number of Unknowns Exceeds Number of Observations

Program exits. This error exit is executed when the program does not have enough good observations, i.e., observations that pass the maximum residual test.

6. Error in Testing Time Points

Program exits. The count of the number of records on the ephemeris tape (Tape 3) is held in location NTAPE3. The value in NTAPE3 is tested before the program tries to find the observation time in the ephemeris. If NTAPE3 ≤ 0 at this time, this error exit occurs.

7. If the value for XNY in the XYZSB Subroutine is zero, the program will exit.

8. Observation Out of Time Range

This is printed when the observation time is not within T_o to T_f. Program continues.

9. Station XXX Not Found

This is printed out when the program is unable to match the station number on the observation card with the station number on the station card.

IV INPUT

1. Specification Cards

For all runs the program requires 5 specification cards.

Cards 1 and 2 — used for headings of output pages.

Card 3 — tells program to start at time 0 minutes and in time steps of Δt, compute the ephemeris for t_f minutes. This card gives diameter (m.) and mass (kg.) of satellite.

Card 4 — This card inputs the elements. A minus sign in column 1 indicates that standard Keplerian elements are being used. Otherwise the elements are taken as those of Reference 1.

Card 5 — Information given on this card controls the type of execution. However, for all runs, the following columns are common:

Col. 1 has code for printed output — Code of 1 prints sub-satellite positions; Code of 2 prints $x, y, z, \dot{x}, \dot{y}, \dot{z}$; Code of 3 prints both; Code of 0 will suppress this printing.

Epoch day given must correspond to that of Keplerian elements. Restart day must be equal to or later than t_f from epoch.

Col. 21 must be 0.

Year of epoch is based on 1958 = 01.

<u>Simulation</u> — Code 0 in column 19.

<u>Differential Correction</u>

Card 5 Col. 20 tells program how many times to repeat correction.

Col. 38 is coded 1 when user wishes to correct both the elements and station positions. (Such stations must have non-zero datum.)

Col. 39 is coded 1 when user wishes to correct the station positions only.

Col. 40 is coded 1 when user wishes to correct the orbital elements only.

The maximum limits for the residuals are left to the descretion of the user, as are the sigmas.

2. Observation Cards

It is preferable that observations be ordered by time. This shortens the running time by eliminating rewind orders.

A card following the last observation card must have "END" punched in Cols. 1-3.

3. Station Cards

The station number must agree in Cols. 1-4 to the station number on the observation card in Cols. 11-14.

The sigmas that may be put in on the station cards are sigmas associated with the observations, not with station position. Observational sigmas on the Observation Cards will take precedence over those on the Station Cards. Also if no sigmas are given on the Station Card, nominal values will be used (ASIG).

4. Options By Means Of Toggle Settings

The program user has options available to him by requesting certain toggles to be set manually by the computer operator at run time. The request is specified on the Request Card and on a card inserted in the data deck immediately following the Job Card, punched as shown below.

Col. 17	Col. 25
HLT	PLEASE PUT ON TOGGLES X, X, etc.

The HLT card causes the message to be printed on the flexo when the run starts.

Toggles 30, 31, 32 and 33 are on for generating simulated observations on punched cards in ANNA format. Any number of these may be specified during a run for the following:

30 for Azimuth and Elevation,

31 for Right Ascension and Declination,

32 for Range,

33 for Range Rate.

Toggle 35 on will cause program to print out Keplerian elements for each time step.

INPUT FORMAT FOR A SIMULATION RUN

May be run with or without station cards.
Without station cards output [illegible] on print option.
With station cards output includes look angles.

CARD 1 — LINE 1 - HEADING FOR OUTPUT PAGES

Col. 1-72 [ ]

CARD 2 — LINE 2 - HEADING FOR OUTPUT PAGES

Col. 1-72 [ ]

CARD 3

Col.	±								±				
1-12	+	0	0	0	0	0	0	0	+	0	0		
13-24													ΔT (minutes)
25-36													Minutes from Epoch to Final Time
37-48													Diameter of Satellite (meters)
49-60													Mass of Satellite (kilograms)

CARD 4

Col.	±								±				
1-12	-												Axis (nautical miles)
13-24													Eccentricity
25-36													Inclination (degrees)
37-48													Mean Anomaly (degrees)
49-60													Arg. of Perigee (degrees)
61-72													Right ascension of node (degrees)

CARD 5

Col.	Field
1	Print Option: 1 gives time, ϕ, λ, h; 2 gives x, y, z, $\dot{x}, \dot{y}, \dot{z}$; 3 gives both
7-10	Epoch Day
11-18	Minutes from Epoch Day to Epoch
19-21	0 0 0
25-28	Restart Day
29-37	Minutes from Restart Day to Restart Time
41-42	Year (1963 = 06)

STATION DATA

	STA #	LATITUDE (F8.5)	LONGITUDE (F9.5)	Alt. meters (F6.0)
Col.	1	5	13	22 27

INPUT FORMAT FOR SPECIFICATION CARDS PROGRAM

Differential Correction

CARD	Columns	Entry	Description
1	9-72		Heading for Output Line 1
2	25-72		Heading (Line 2)

CARD	Columns	Entry	Description
3*	1-12	+0 0 0 0 0 0 0 0 + 0 0	T_o
	13-24		ΔT (minutes)
	25-36		Time from Epoch to Final Time (minutes)
	37-48		Diameter of Vehicle (meters)
	49-60		Mass of Vehicle (kilograms)

OPTION - Use if only Keplerian elements are available; code with a Minus (-) in Col. 1.

CARD	Columns	Entry	Description
4a*	1-12	-	Axis (nautical miles)
	13-24		Eccentricity
	25-36		Inclination (degrees)
	37-48		Mean Anomaly (degrees)
	49-60		Arg. of Perigee (degrees)
	61-72		Rt Asc of Node (degrees)

OPTION - Use if the following values are supplies as input; otherwise, see 4a.

CARD	Columns	Entry	Description
4b*	1-12		L_o (radians)
	13-24		Axn
	25-36		Ayn
	37-48		Hx Earth radii
	49-60		Hy " "
	61-72		Hz " "

CARD	Columns	Entry	Description
5	1-6		Print Option (1 for Time, ϕ, λ, ht; 2 for Time, x, y, z, $\dot{x}$, $\dot{y}$, $\dot{z}$; 3 for both)
	7-10		Epoch (Day of Year; integer)
	11-18		Time; from beginning of Epoch Day to Epoch - minutes)
	19		Blank for differential correction; 0 (zero) for simulation
	20		Number of times to repeat correction
	21	0	
	22-24		Sigma 1
	25-28		Restart Time (day of year)
	29-37		Restart Time (minutes)
	38		Code 1 to correct both elements and station positions. Use 1 of 3 only
	39		Code 1 to correct station positions only
	40		Code 1 to correct orbital elements only
	41-42		Year - 1958 = 01; 1959 = 02; 1960 = 03; 1961 = 04; 1962 = 05; 1963 = 06; etc
	43-48		Max. for angle and range residuals
	49-51		Sigma 2
	52-54		Sigma 3
	55-57		Sigma 4
	58-60		Sigma 5
	61-63		Sigma 6
	64-66		Sigma 7
	67-69		Sigma 8
	70-72		Sigma 9
	73-78		Max. for range rate residuals

*Format (±xxxxxxxx±xx)

Input Format for OPTICAL Observations

* Code 2 identifies observation as Optical

** Sigmas have decimal point assumed to be between col. 60 & 61 and col. 64 & 65

				OBSERVATION TIME							RT ASC (- in Col. 30) hrs / AZIMUTH				DECL. / ELEVATION				Sigma**	Sigma**
YR	SAT. I. D.	*	STATION I. D.	yr	mo	day	hr	min	sec	.0001 sec	deg	min	sec	.01 sec	deg	min	sec	.1 sec	σ or A	δ or h
1	3	6	11	15	16	18	20	22	24	26 29	30	33	35	37	39	43	45	47	59 60 61 62	63 64 65 66

Input Format for RANGE RATE Observations

* Code 4 identifies data as a Base Frequency
Code 3 identifies data as a Received Frequency
** Sigma has decimal point assumed to be between Col. 50 and 51

YR	SAT I.D.	*	STATION I.D.	OBSERVATION TIME							BASE OR REC'D FREQUENCY cycles per second	Sigma cps**
				yr	mo	day	hr	min	sec	.0001 sec		
1	3	6	11	15							38	49

Input Format for RANGE Observations

* Code 1 identifies the type of observation as range

** The sigma for range is given in col 49-51 with decimal point assumed between col 50&51

	YR	SAT I. D.	*	STATION I. D.	OBSERVATION TIME							Format ± xxxxxxxxxx±yy	Sigma
					yr	mo	day	hr	min	sec	.0001 sec	RANGE (meters)	**
Col	1	3	6	11	15	16	18	20	22	24	26 29	35 (rt. adjusted) 48	49

Input Format - STATION Cards

						ρ_{obs}	$\dot{\rho}_{obs}$	α or A	δ or h
	SAT I.D.	LATITUDE (degrees)	LONGITUDE (degrees)	HEIGHT (meters)	Datum no.	ASIG 1	ASIG 2	ASIG 3	ASIG 4
COL	1–4	5–12 (F8.5)	13–21 (F9.5)	22–27 (F6.0)	33-36	38–45	46–53	54–61	62–69

V COMPILATION PROCEDURE

The program must be compiled on the Philco "2000" under the system called "Altac 2". This system requires that all tape units used be specified according to their use in an "Iounits" statement.

Decimal Input Tapes are LTO and LT10
Intermediate Binary Tapes are LT3, LT6, LT9, LT11,
LT13, LT14, and LT15
Decimal Output Tape is LT5, Data Select 0
Punched Card Output is LT13, Data Select 10

The program uses approximately 100 RPL block. Compilation time is about 20 minutes. The source deck consists of approximately 5000 cards (2 1/2 boxes).

VI EXECUTION OF DATA RUN

Col.	Col.
17	25
JOB	XXX (Identify run)
REWIND	0, 3, 6, 9, 10, 11, 13
RPL	1, DATA, GO
TAPE	0

For all runs, insert 5 specification or control cards here. For differential correction runs, the 5 specification cards are followed by the observation cards. The last observation card must be followed by a card with "END" in Cols. 1-3.

ENDDATA	
REWIND	0
JMP	*
TAPE	10

Station cards are inserted here. The run must have a card with "END" in Cols. 1-3 before the following:

ENDDATA	
REWIND	10
JPL	4, XXXX, GO (ID of program)

TAPE UNITS Requiring Write Rings

1. Input

   TAPE 0 — 5 specification cards and observation cards.

   TAPE 10 — station cards

2. Output

   TAPE 5 — printed output

   TAPE 13 — punched cards using data select 10 when making a simulation run.

3. Binary

TAPE 3	saves time and sub-satellite positions
TAPE 6	saves time, x, y, z, $\dot{x}$, $\dot{y}$, $\dot{z}$.
TAPE 9	saves observations after processing
TAPE 11	saves data used in computing sigmas
TAPE 14	saves data used in computing the corrected latitude, longitude, and height of the observing station (spheroid specified).
TAPE 15	saves elements at every time step if Toggle switch 35 is on.

VII OUTPUT

A. Printed Output:

1. Input data from the 5 specification cards.

2. L_o, A_{xn}, A_{yn}, H_x, H_y, H_z if input is in form of Keplerian elements.

3. Time and satellite position in latitude, longitude, and altitude above a specified reference spheroid for each integration step. This is available with a print option of 1 or 3.

4. Time and inertial position and velocity $(x, y, z, \dot{x}, \dot{y}, \dot{z})$ of satellite for each integration step. This is available with a print option of 2 or 3.

5. Input data from station cards.

6. Input data from observation cards.

7. Station number, time, and residual for each observation.

8. Root-mean-square of the non-rejected residuals (SUM and SUM2) and the corrections to the orbital parameters (DELTA A/A, DELTA AXN, DELTA AYN, DELTA L_o, DELTA NODE, DELTA I). The SUM2 column contains range rate residuals.

9. Suggested changes in coordinates of datum in earth-fixed coordinates (DELTA X, DELTA Y, and DELTA Z).

10. The differentially corrected Keplerian elements, differentially corrected L_o, A_{xn}, A_{yn}, H_x, H_y, H_z, and the corrected x, y, z, $\dot{x}$, $\dot{y}$, $\dot{z}$ at epoch.

11. New positions for stations in non-zero datums, station number, latitude, longitude, elevation, and earth fixed rectangular coordinates (x, y, z) of station. Latitude, longitude and elevation are referenced to the spheroid whose parameters are specified in the Table of Program Constants.

12. The standard deviation (sigma) for DELTA A/A, DELTA AXN, DELTA AYN, DELTA L_0, DELTA NODE, DELTA I at epoch.

13. The standard deviation (sigma) in the observation station position (SIGMA X, SIGMA Y, and SIGMA Z).

14. Time and the satellite position (x, y, z) with the associated standard deviation are given at epoch and for each observation time.

B. Punched Card Output

An optional feature of DOC II will produce observation cards punched in correct format for use as input in the differential correction portion of the program.

These cards can be generated during a simulation run by toggle settings on the console. The settings are made manually by the computer operator as specified by the program user on the Computer Request Card, and by the insertion of a card in the run deck, immediately following the job card, as follows:

(Col. 17) HLT (Col. 25) PUT ON TOGGLES X, X, ETC.

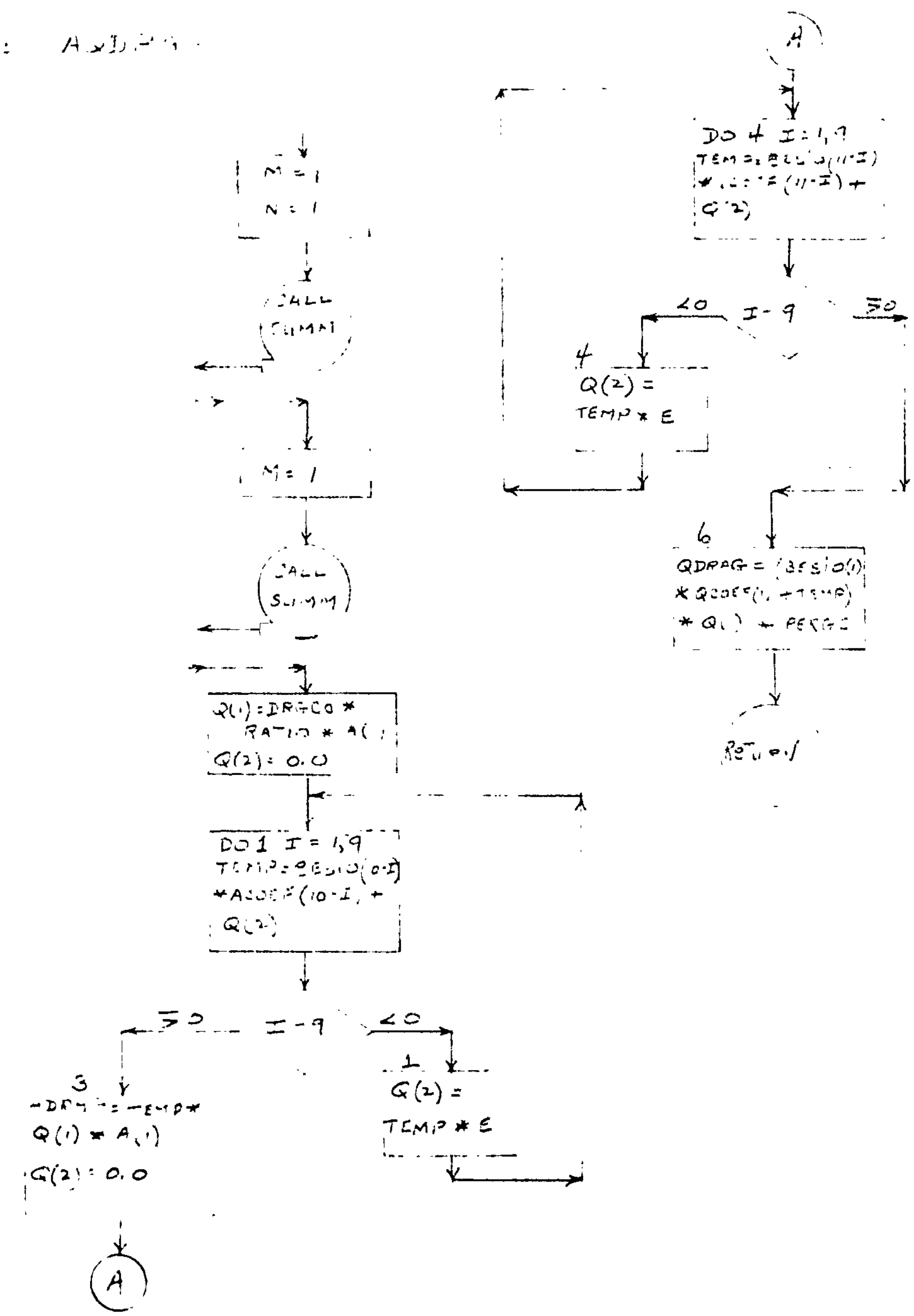
This card will be printed out on the flexo-writer at run time for instruction to the operator.

Set Toggle 30 for azimuth and elevation observations
Set Toggle 31 for right ascension and declination observations
Set Toggle 32 for range observations
Set Toggle 33 for range rate observations

VIII

FLOW CHART OF SUBROUTINES

Subroutine



M = 1
N = 1
CALL SUMM
M = 1
Q(2) = 0.0
DO 1 I = 1,9
I - 9
<0
1
Q(2) =
TEMP * E
3
Q(2) = 0.0
A
DO 4 I = 1,9
I - 9
<0
4
Q(2) =
TEMP * E
6

Subroutine BELGE

```
INITIALIZE
ALFA (K, I) and
BETA (K, I)
K = 1, 6
J = 1, 6
```

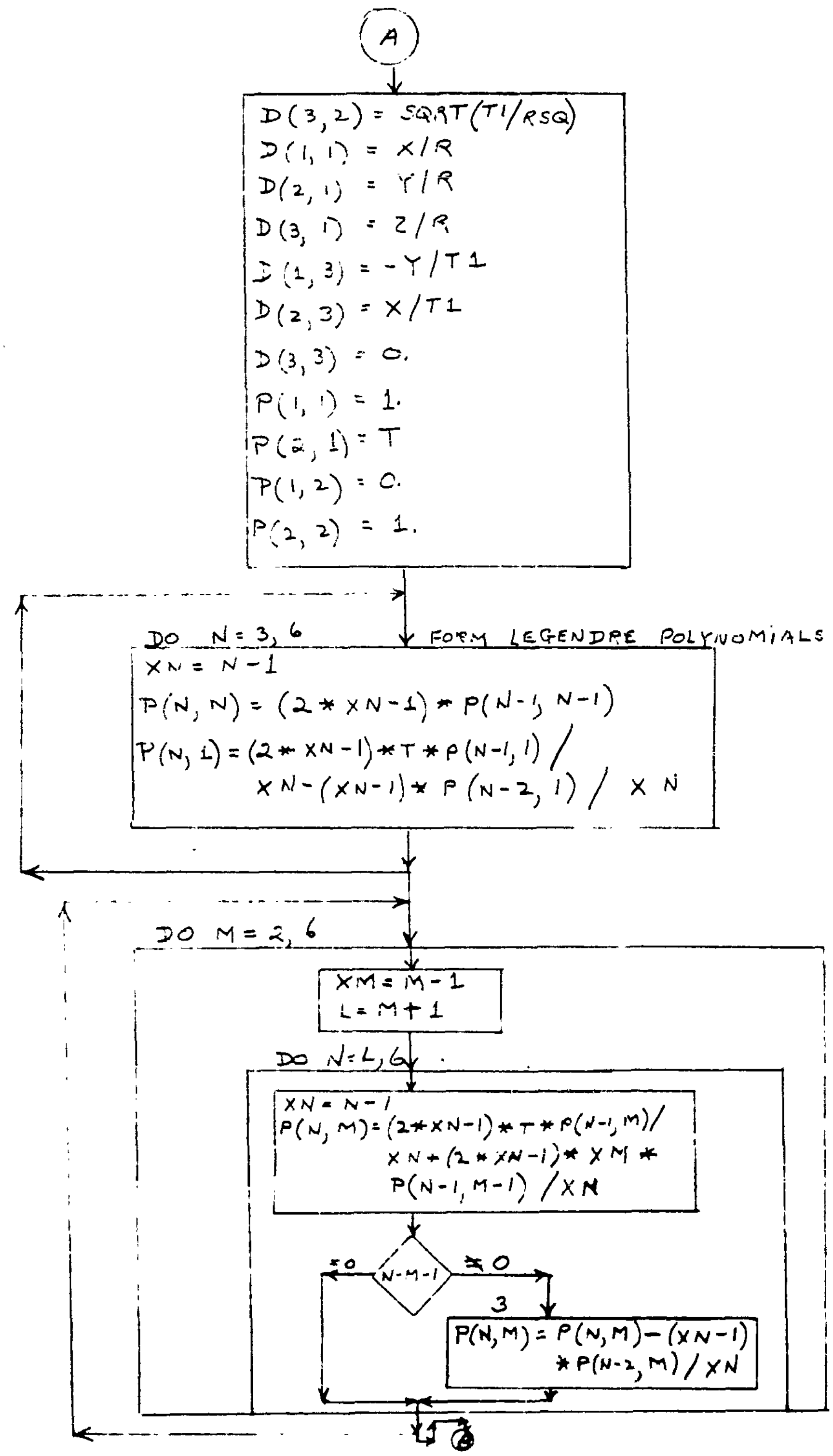
```
ALFA(3,3) = 1.056 E-6 ; BETA (3,3) = .799 E-6
ALFA (4,2) = 1.26 E-6; BETA (4,2) = .45 E-6
ALFA (4,3) = .210 E-6; BETA (4,3) = .0 6 E-6
ALFA (4,4) = .121 E-6; BETA (4,4) = .65 E-6
ALFA(5,2) = .405 E-6; BETA (5,2) = .32 E-6
ALFA (5,3) = .0793 E-6; BETA (5,3) = .13 E-6
ALFA (5,4) = .069 E-6; BETA (5,4) = .0  4 E-6
ALFA (5,5) = .0058 E-6; BETA (5,5) = .012 E-6
```

20

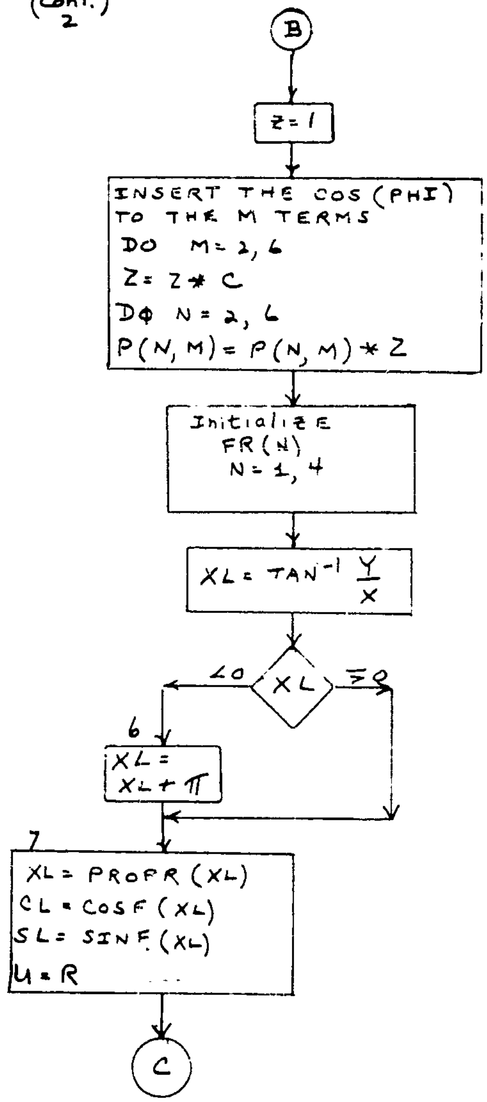
```
COMPUTE
CT = COSF(TH) ; ST = SINF(TH)
X = W( ) * CT + W(2) * ST
Y = - W( ) * ST + W(2) * CT
Z = W(3)
T1 = X^2 + Y^2
QMU = 1.0
RSQ = T1 + Z^2
T2 = Z/SQRTF(T1)
D(1,2) = - T2 * X /RSQ
D(2,2) = - T2 * Y/RSQ
R = √RSQ ; F = ATANF(T2)
C = COS(F); T = SIN(F)
```

A

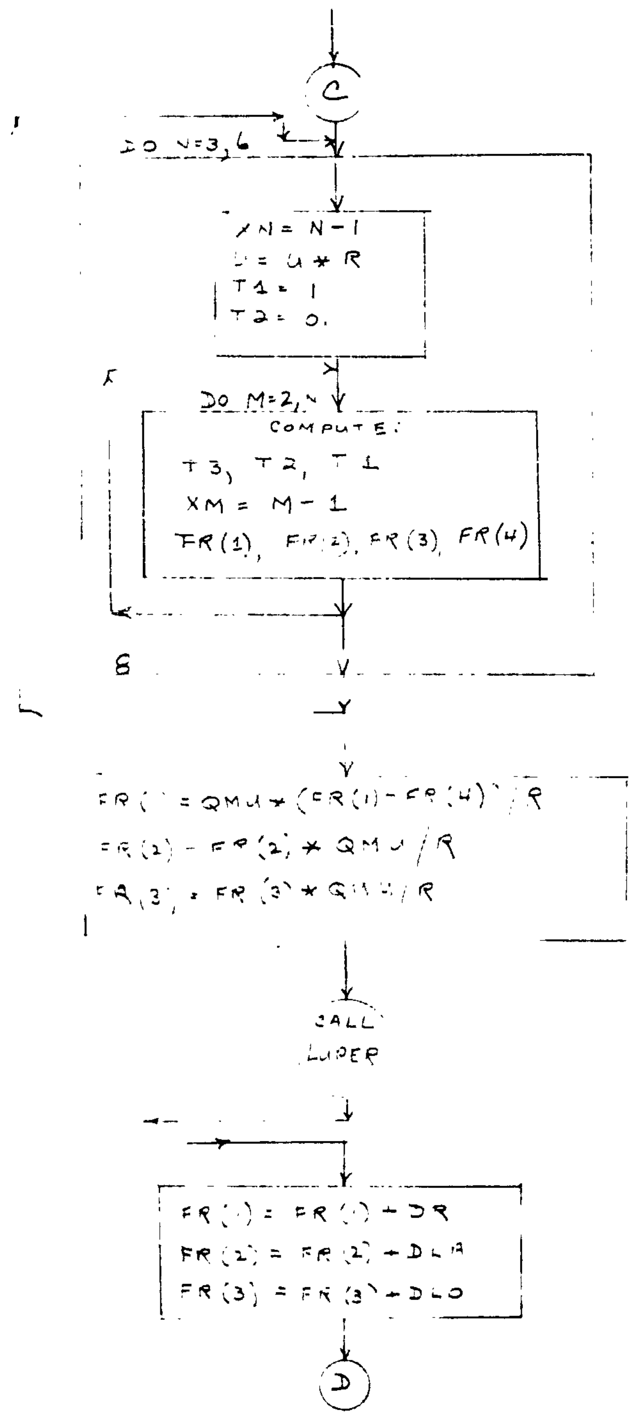
BELGE (CONT) (1)


A
D(3,2) = SQRT(T1/RSQ)
D(1,1) = X/R
D(2,1) = Y/R
D(3,1) = Z/R
D(1,3) = -Y/T1
D(2,3) = X/T1
D(3,3) = 0.
P(1,1) = 1.
P(2,1) = T
P(1,2) = 0.
P(2,2) = 1.
DO N = 3, 6
FORM LEGENDRE POLYNOMIALS
XN = N-1
P(N,N) = (2*XN-1)*P(N-1,N-1)
P(N,1) = (2*XN-1)*T*P(N-1,1)/XN-(XN-1)*P(N-2,1)/XN
DO M = 2, 6
XM = M-1
L = M+1
DO N=L,6
XN = N-1
P(N,M) = (2*XN-1)*T*P(N-1,M)/XN+(2*XN-1)*XM*P(N-1,M-1)/XN
N-M-1
=0
≠0
3
P(N,M) = P(N,M)-(XN-1)*P(N-2,M)/XN
B

BELGE (Cont. 2)

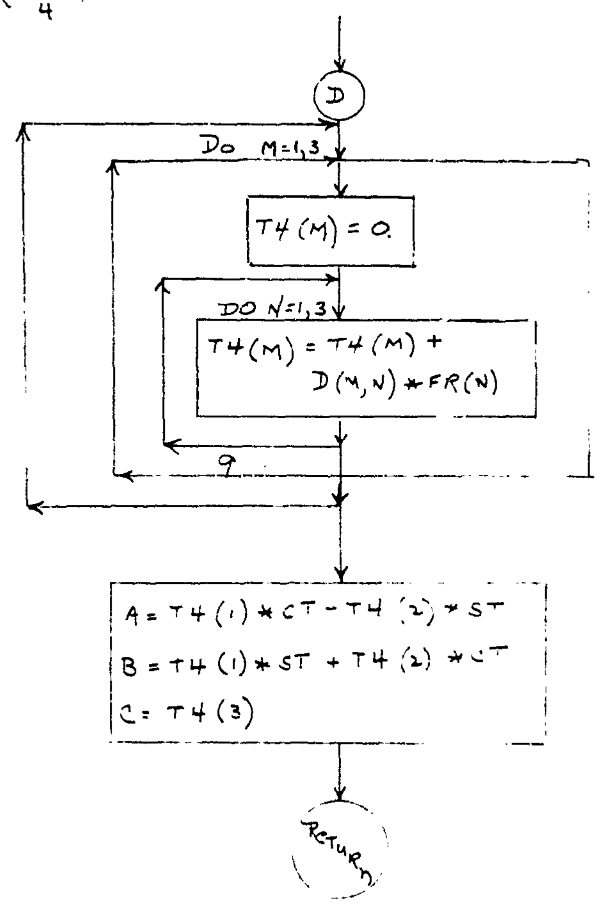

B
Z = 1
INSERT THE COS (PHI) TO THE M TERMS
DO M = 2, 6
Z = Z * C
DΦ N = 2, 6
P(N, M) = P(N, M) * Z
Initialize FR(N) N = 1, 4
XL = TAN^-1 Y/X
XL
<0
≥0
6
XL = XL + π
7
XL = PROPR (XL)
CL = COSF (XL)
SL = SINF (XL)
U = R
C

BELGE (cont 3)

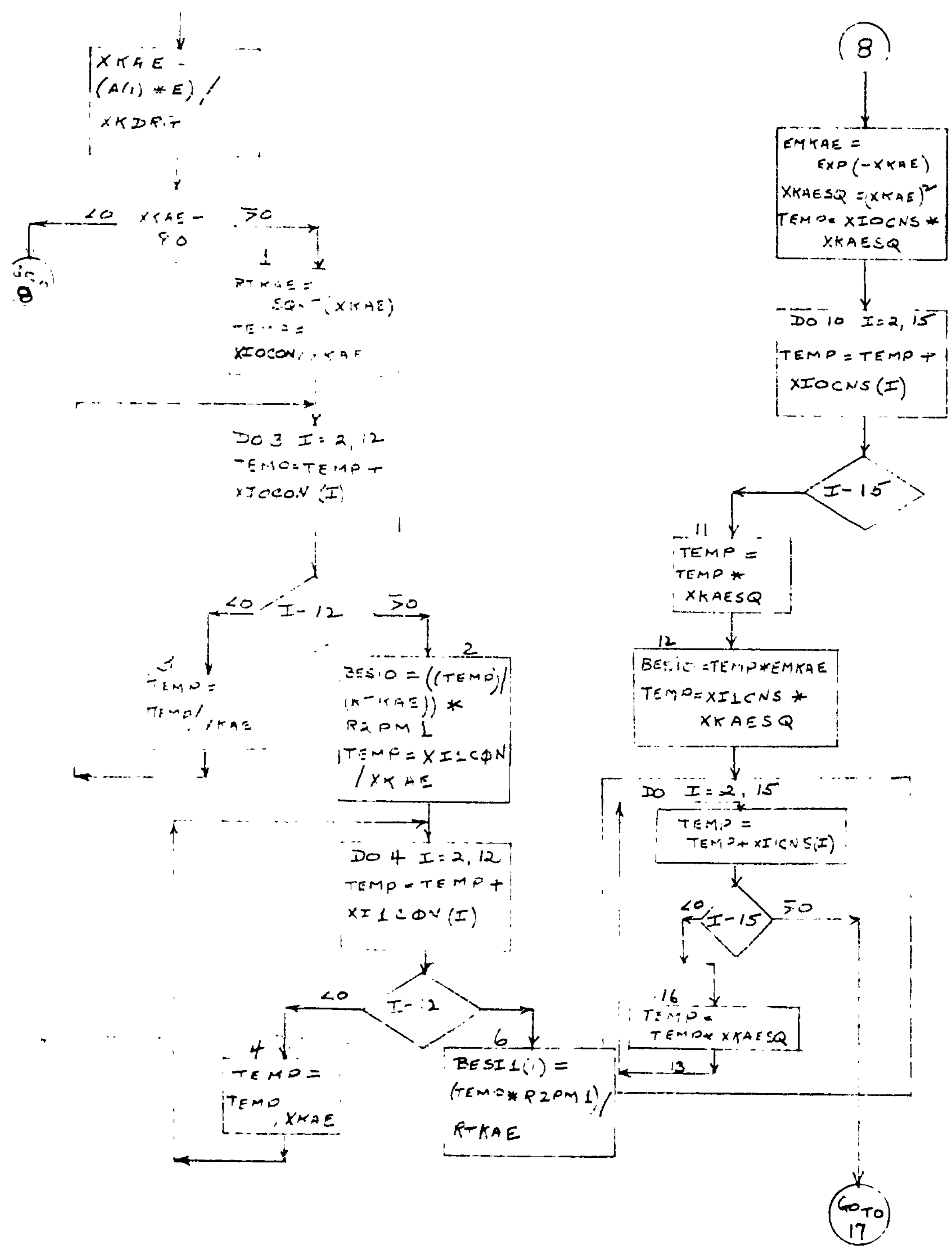


C
DO N=3,6
XN = N-1
U = U * R
T1 = 1
T2 = 0.
DO M=2,N
COMPUTE:
T3, T2, T1
XM = M-1
FR(1), FR(2), FR(3), FR(4)
8
FR(1) = QMU * (FR(1)-FR(4))/R
FR(2) = FR(2) * QMU/R
FR(3) = FR(3) * QMU/R
CALL LUPER
FR(1) = FR(1) + DR
FR(2) = FR(2) + DLA
FR(3) = FR(3) + DLO
D

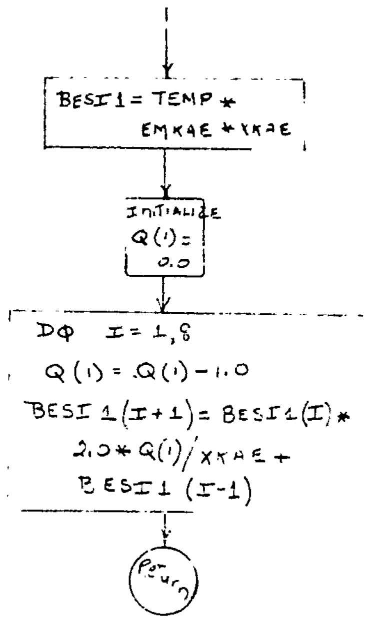
BELGE (cont. 4)


D
Do M=1,3
T4(M) = 0.
DO N=1,3
T4(M) = T4(M) + D(M,N) * FR(N)
9
A = T4(1) * CT - T4(2) * ST
B = T4(1) * ST + T4(2) * CT
C = T4(3)
RETURN

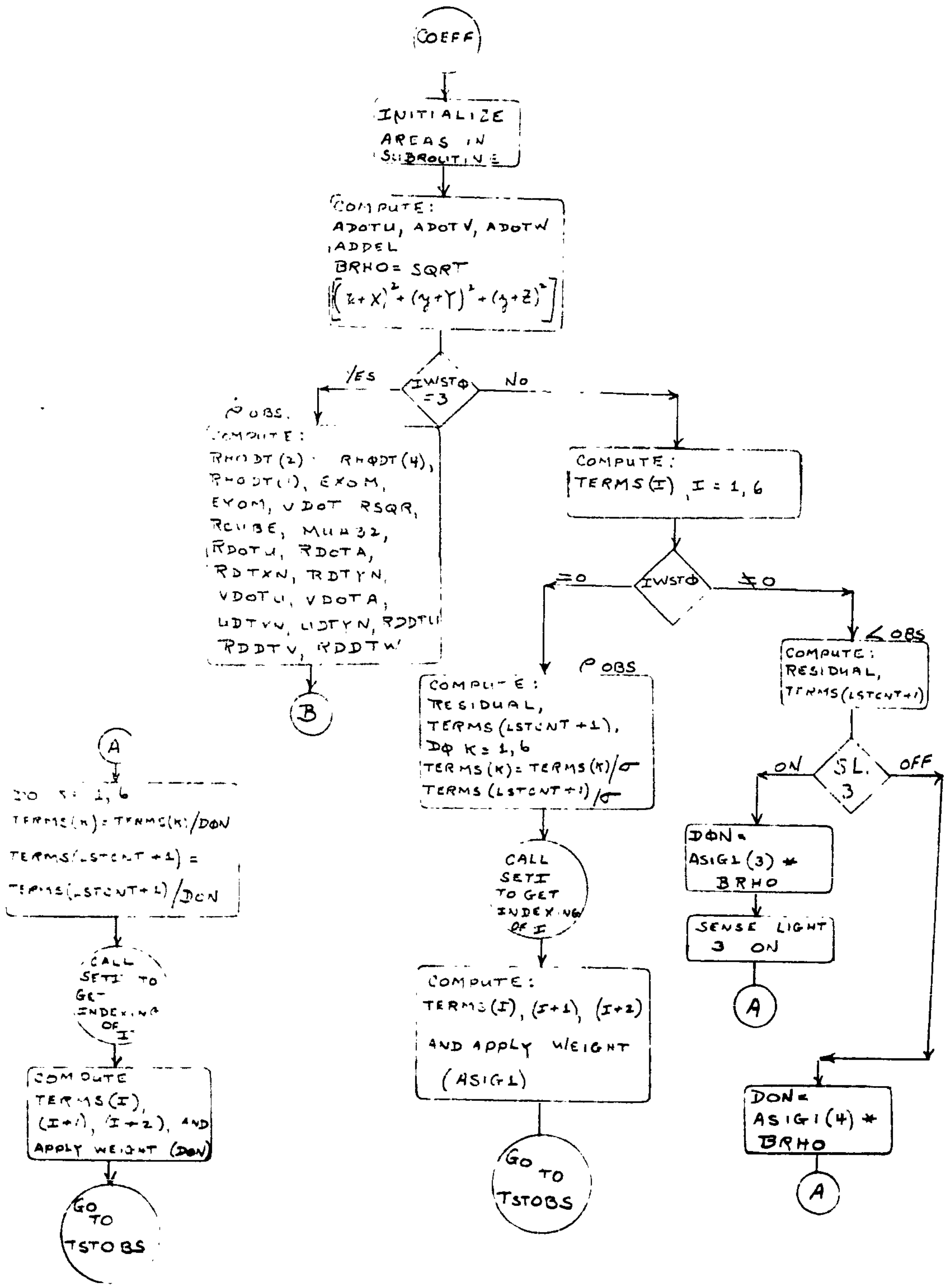
Subroutine BESSEL

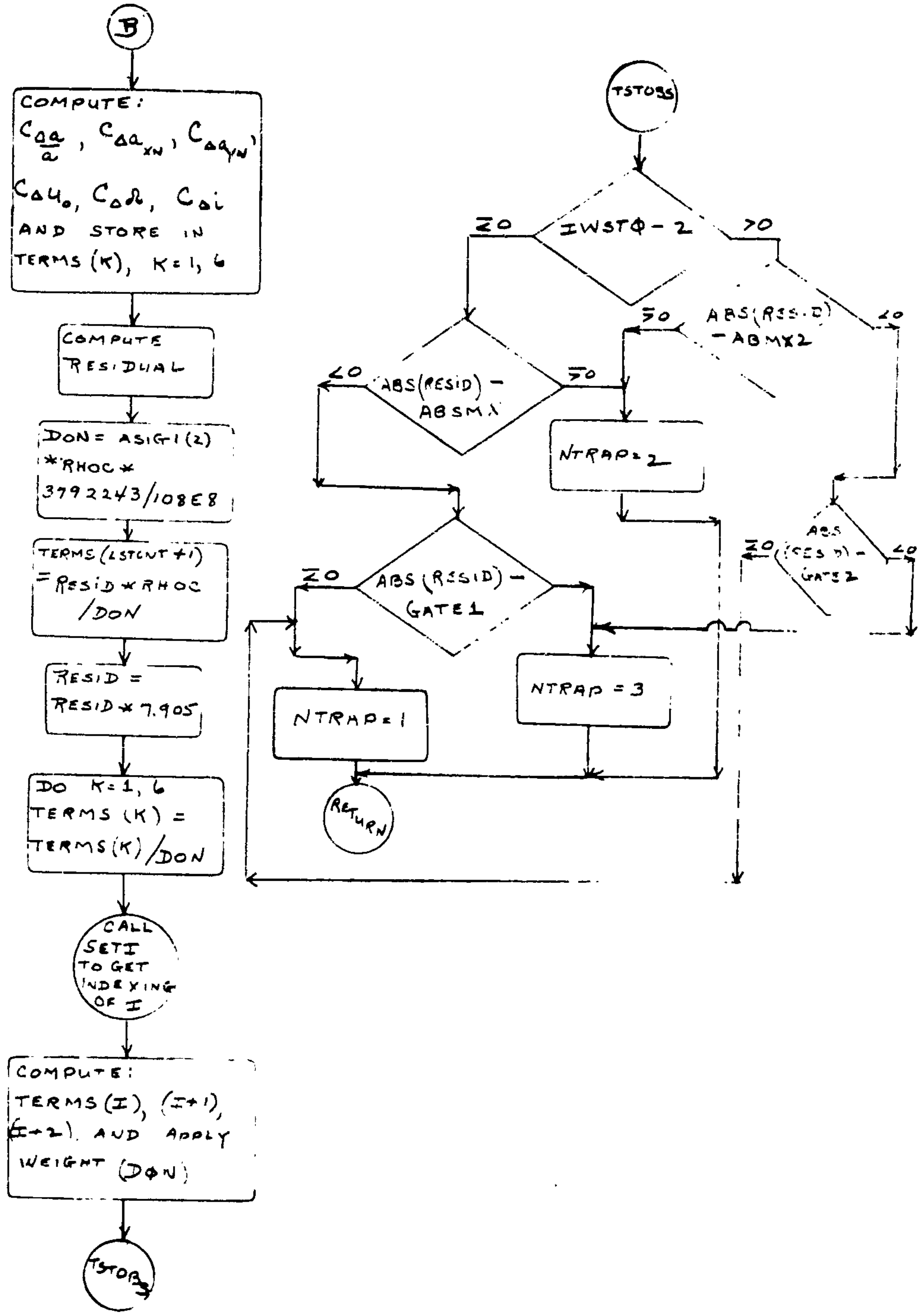


BESSEL (cont.)

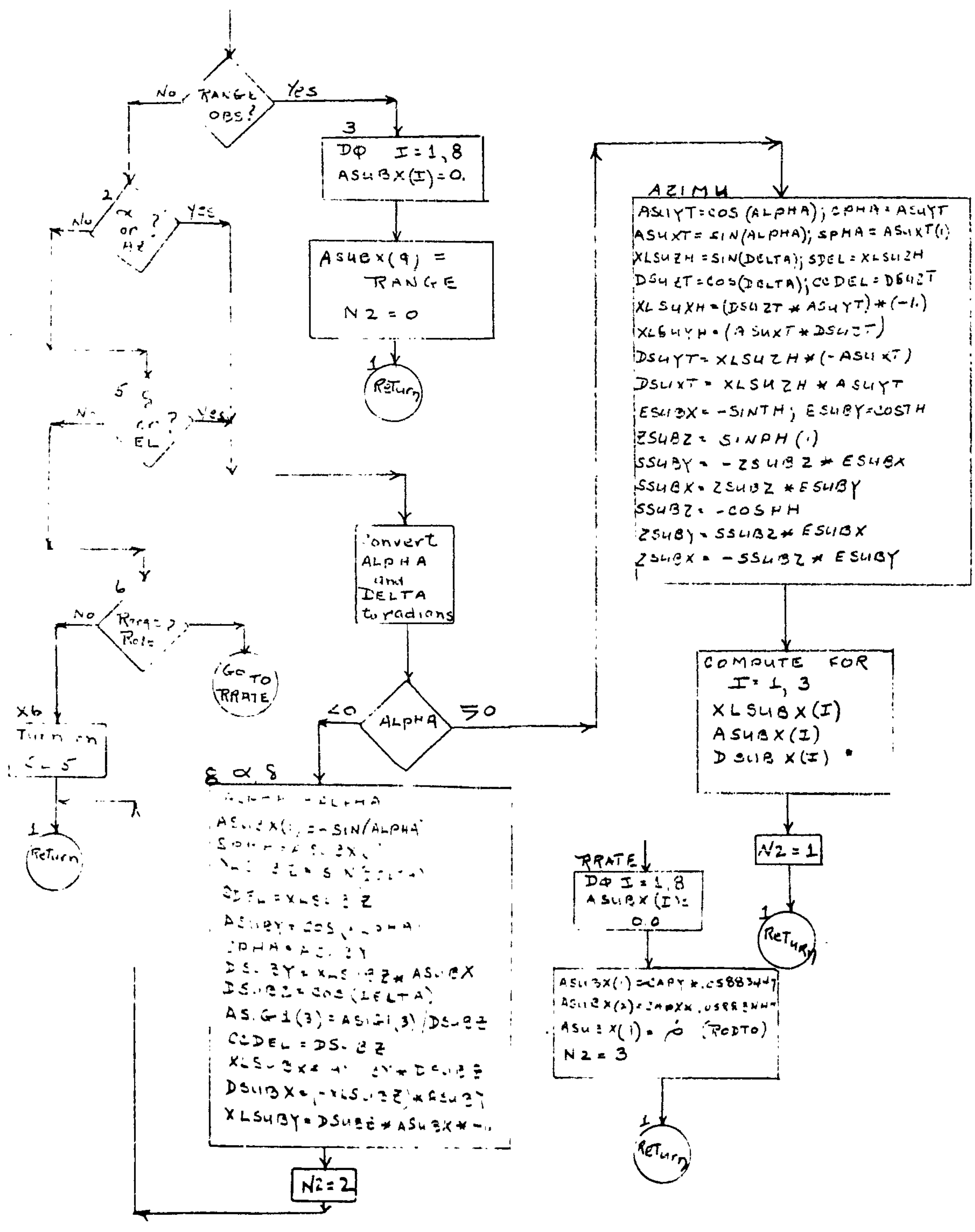

BESI1 = TEMP * EMKAE * XKAE
INITIALIZE Q(1) = 0.0
DΦ I = 1, 8
Q(1) = Q(1) - 1.0
BESI1(I+1) = BESI1(I) * 2.0 * Q(1)/XKAE + BESI1(I-1)
Return

SUBROUTINE COEFF

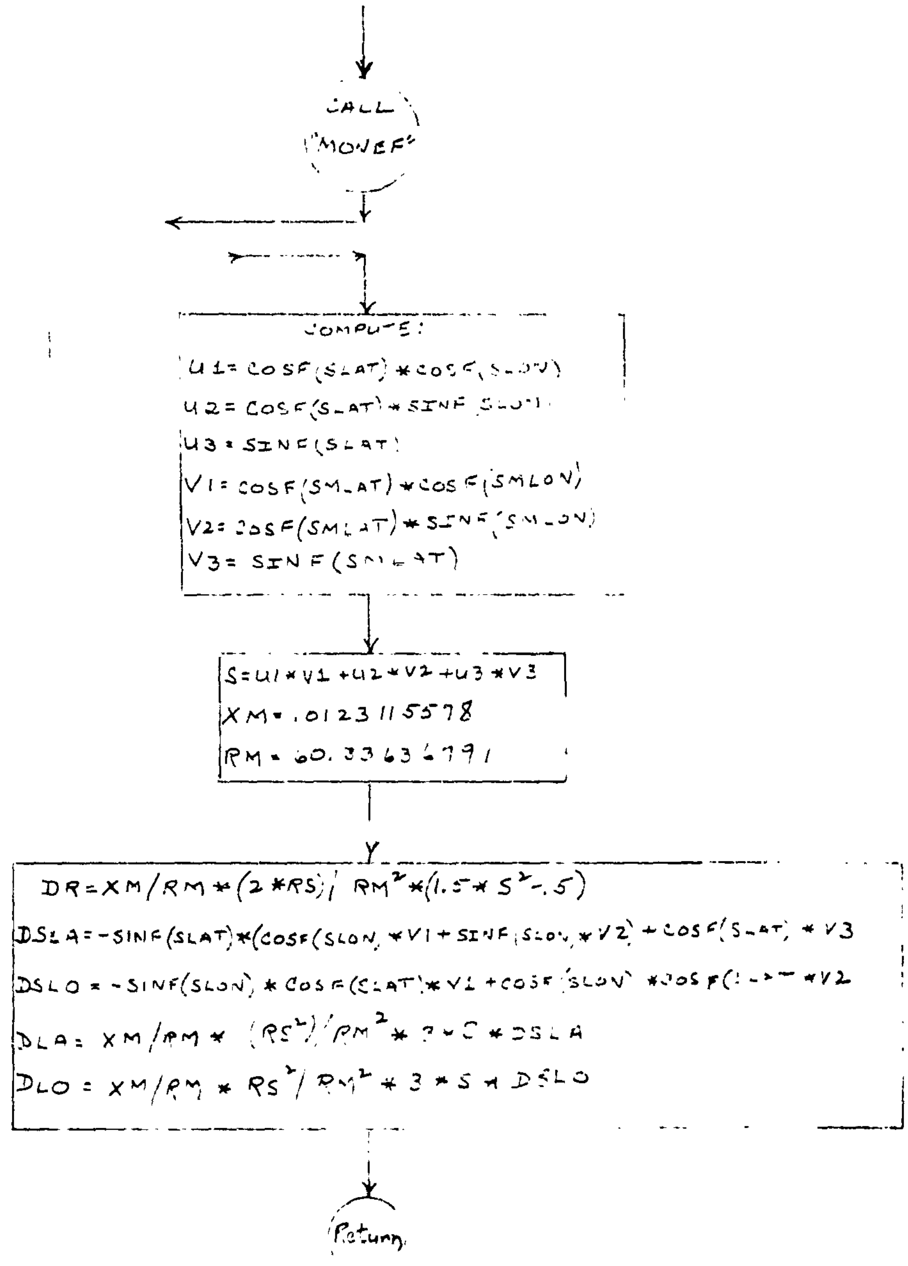



B
COMPUTE:
$C_{\frac{\Delta a}{a}}$, $C_{\Delta a_{XN}}$, $C_{\Delta a_{YN}}$,
$C_{\Delta u_0}$, $C_{\Delta \Omega}$, $C_{\Delta i}$
AND STORE IN
TERMS (K), K = 1, 6
COMPUTE RESIDUAL
DON = ASIG1 (2) * RHOC * 3792243/108E8
TERMS (LSTCNT + 1) = RESID * RHOC /DON
RESID = RESID * 7.905
DO K = 1, 6
TERMS (K) = TERMS (K)/DON
CALL SETI TO GET INDEXING OF I
COMPUTE:
TERMS (I), (I+1), (I+2) AND APPLY WEIGHT (DØN)
TSTOB3
TSTOB3
IWSTØ - 2
≥0
>0
ABS(RESID) - ABSMX
<0
≥0
ABS(RESID) - ABMX2
≥0
<0
NTRAP = 2
ABS (RESID) - GATE 1
≥0
NTRAP = 1
NTRAP = 3
RETURN
ABS (RESID) - GATE 2
≥0
<0

Subroutine GTOBS

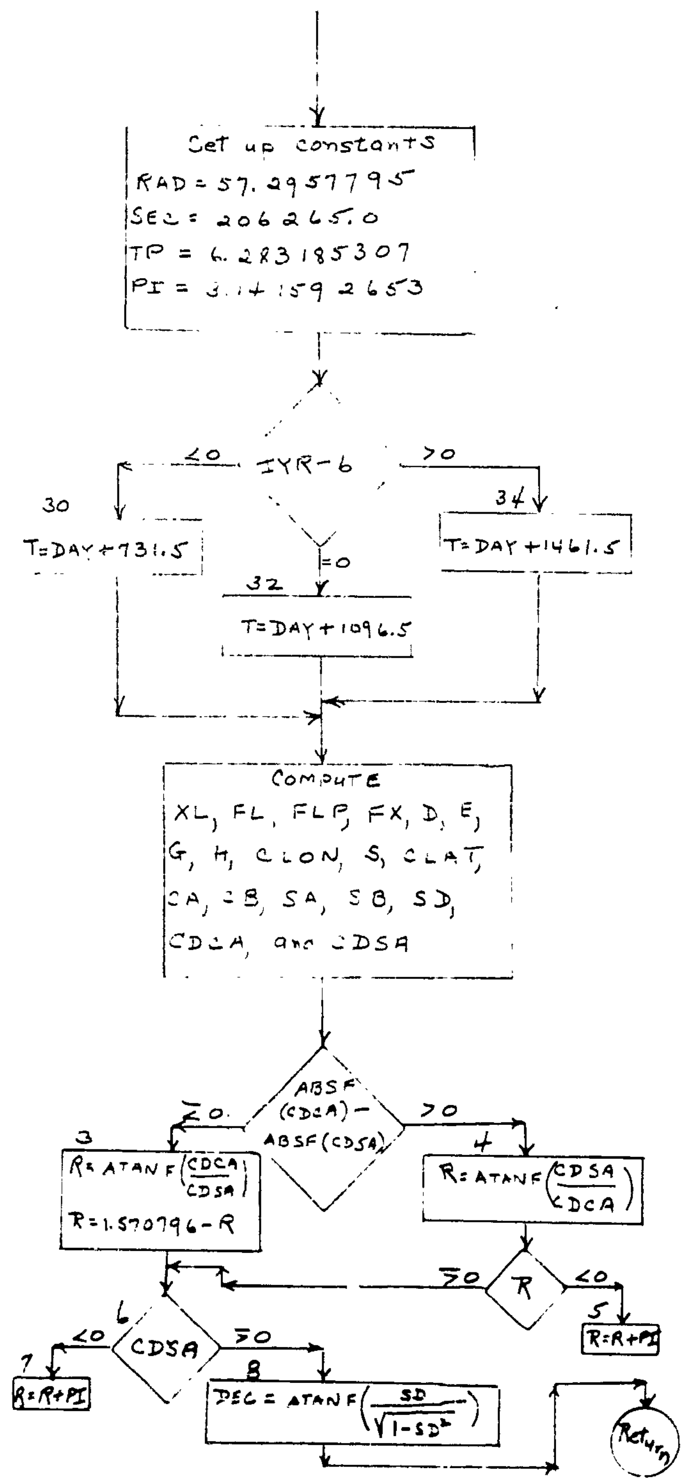


Subroutine [illegible]



CALL
MONEF
COMPUTE:
U1= COSF(SLAT)*COSF(SLON)
U2= COSF(SLAT)*SINF(SLON)
U3= SINF(SLAT)
V1= COSF(SMLAT)*COSF(SMLON)
V2= COSF(SMLAT)*SINF(SMLON)
V3= SINF(SMLAT)
S=U1*V1+U2*V2+U3*V3
XM=.0123115578
RM=60.3363[illegible]791
DR=XM/RM*(2*RS)/RM²*(1.5*S²-.5)
DSLA=-SINF(SLAT)*(COSF(SLON)*V1+SINF(SLON)*V2)+COSF(SLAT)*V3
DSLO=-SINF(SLON)*COSF(SLAT)*V1+COSF(SLON)*COSF([illegible]*V2
DLA= XM/RM*(RS²)/RM²*3*S*DSLA
DLO= XM/RM*RS²/RM²*3*S*DSLO
Return

Subroutine MONEF



Subroutine PRESS

Set up constants

C1 = .9856472

P1 = .456E-6

P2 = 1.006814784

PI = 3.14159265 3

HRK = 0.0

Set I = 1 for 1962

I = 2 for 1963

I = 3 for 1964

COMPUTE:

SOL = SUNL(I) + C1*T + C15(I) * SINF(C1*T − C14(I))

RA = SOL − C16(I) * SINF(SOL + SOL)

DEC = ATANF(P2 * C17(I) * SINF(RA))

CRA = COSF(RA)

CDEC = COSF(DEC)

SRA = SINF(RA)

SDEC = SINF(DEC)

COMPUTE:

U1 = CRA * CDEC

U2 = SRA * CDEC

U3 = CDEC

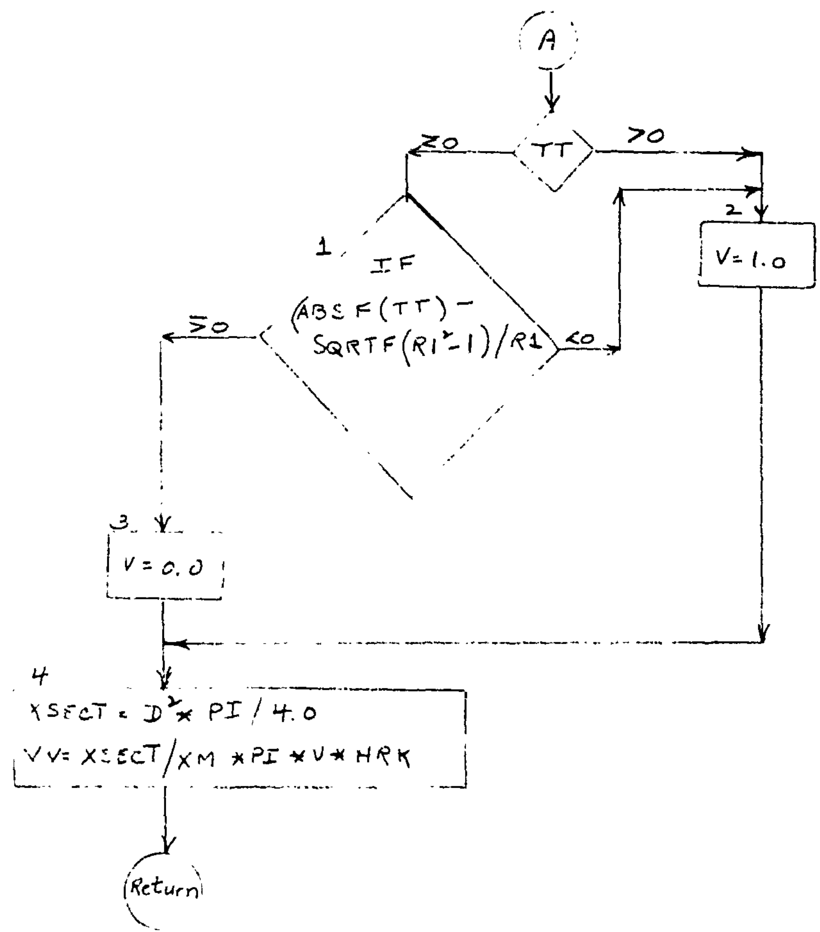
$R1 = \sqrt{X(1)^2 + X(2)^2 + X(3)^2}$

V1 = X(1)/R1; V2 = X(2)/R1; V3 = X(3)/R1

TT = U1 * V1 + U2 * V2 + U3 * V3

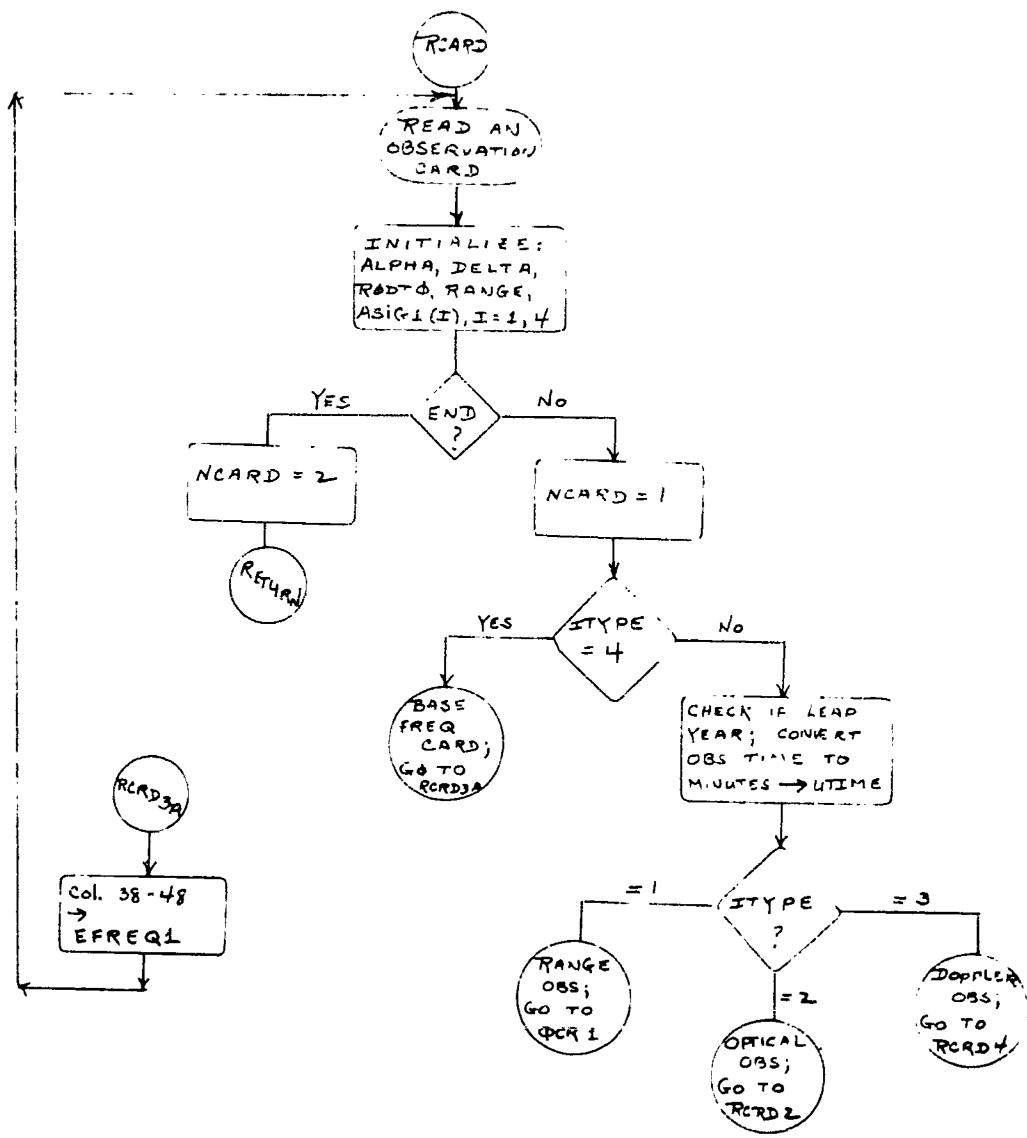
A

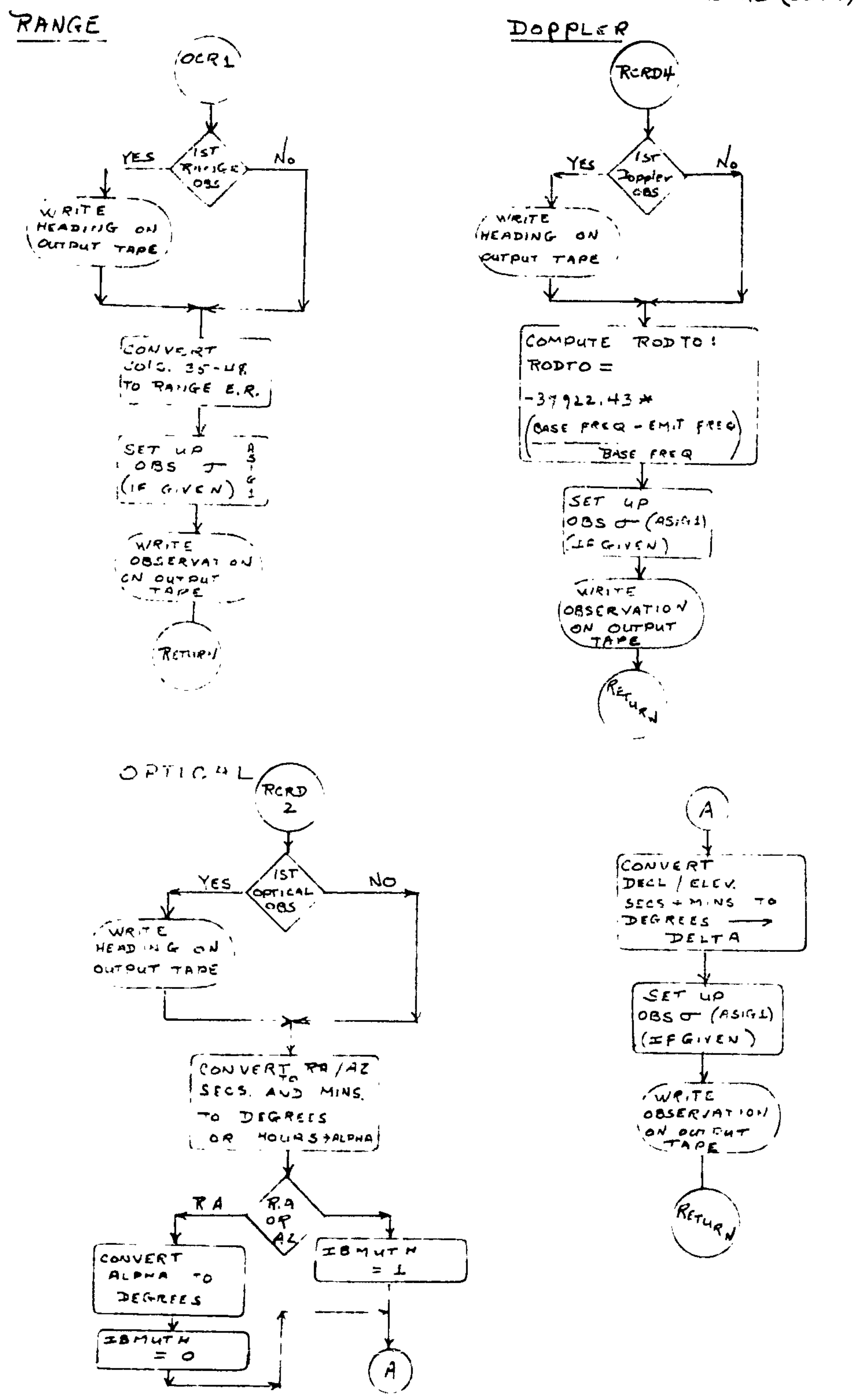
PRESS (cont)


A
TT
≥0
>0
1
IF
(ABSF(TT) − SQRTF(R1²−1)/R1
≥0
<0
2
V=1.0
3
V=0.0
4
XSECT = D² * PI / 4.0
VV= XSECT/XM * PI * V * HRK
Return

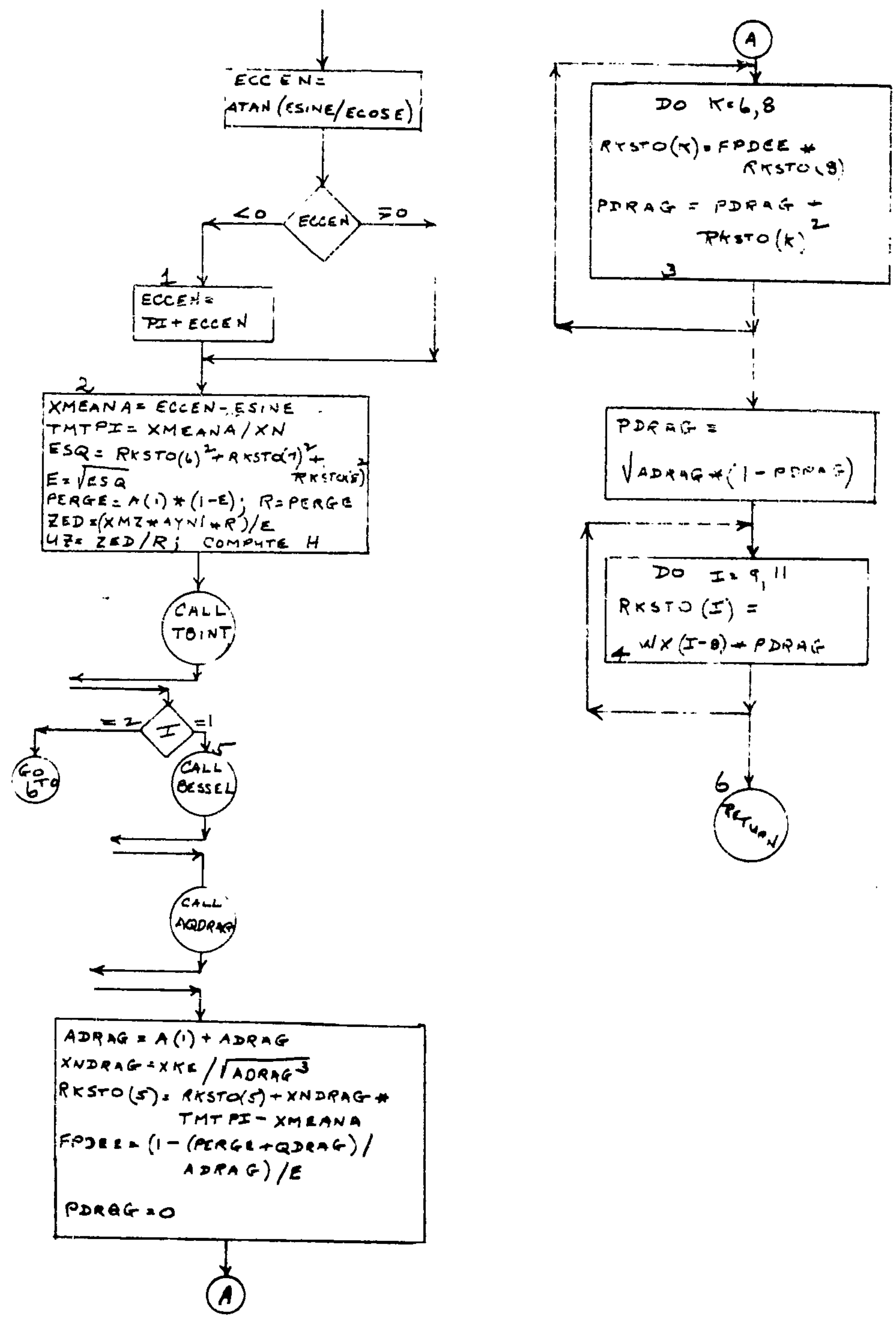
SUBROUTINE: RCARD

Read and convert an observation card



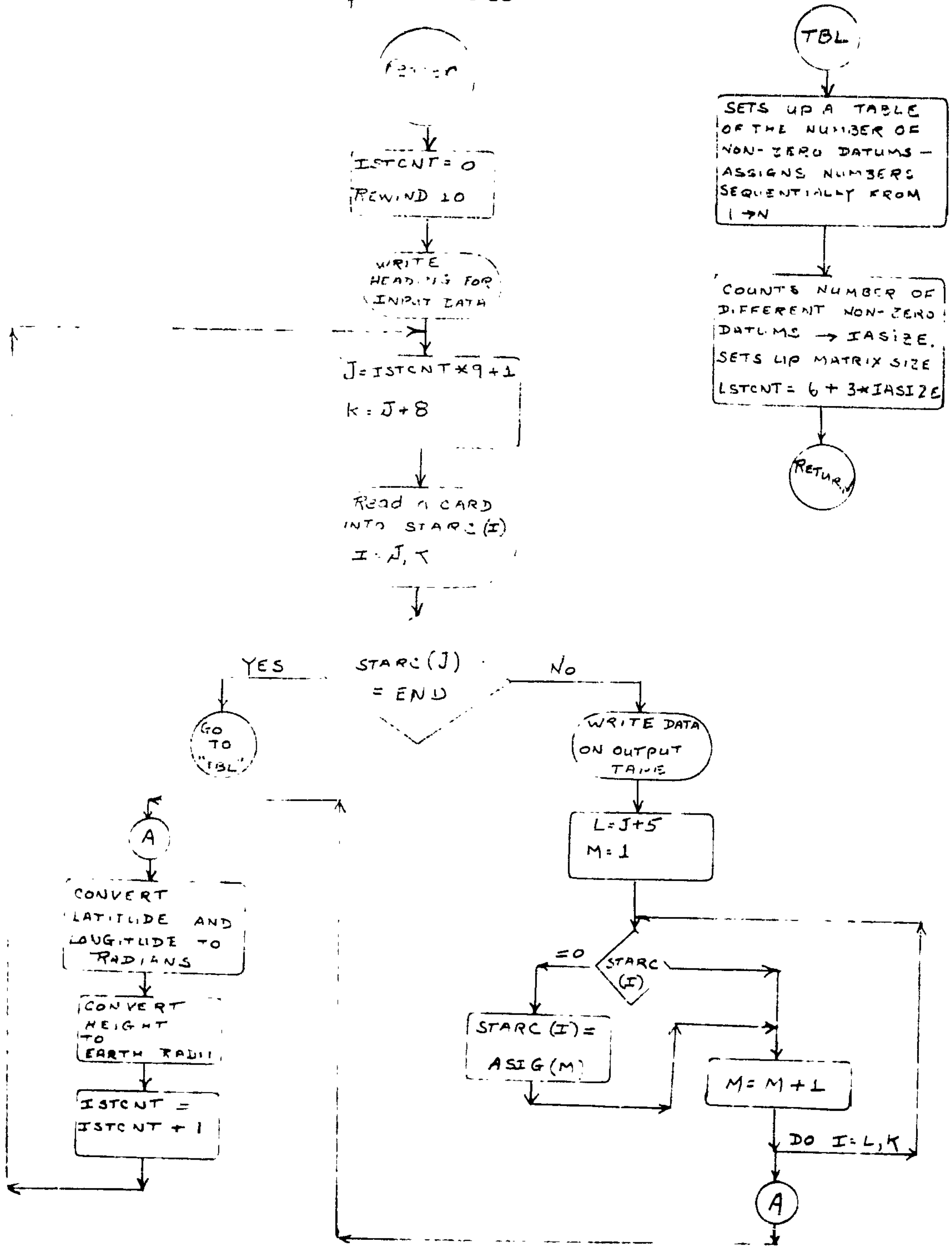

RANGE
OCR1
1ST RANGE OBS
YES
NO
WRITE HEADING ON OUTPUT TAPE
CONVERT COLS. 35-48 TO RANGE E.R.
SET UP OBS σ (IF GIVEN) ASIG1
WRITE OBSERVATION ON OUTPUT TAPE
RETURN
DOPPLER
RCRD4
1ST Doppler OBS
YES
NO
WRITE HEADING ON OUTPUT TAPE
COMPUTE RODTO: RODTO = -37922.43 * (BASE FREQ - EMIT FREQ / BASE FREQ)
SET UP OBS σ (ASIG1) (IF GIVEN)
WRITE OBSERVATION ON OUTPUT TAPE
RETURN
OPTICAL
RCRD 2
1ST OPTICAL OBS
YES
NO
WRITE HEADING ON OUTPUT TAPE
CONVERT RA/AZ SECS. AND MINS. TO DEGREES OR HOURS + ALPHA
RA OR AZ
RA
CONVERT ALPHA TO DEGREES
IBMUTH = 0
IBMUTH = 1
A
CONVERT DECL / ELEV. SECS + MINS TO DEGREES → DELTA
SET UP OBS σ (ASIG1) (IF GIVEN)
WRITE OBSERVATION ON OUTPUT TAPE
RETURN

Subroutine RCTFY

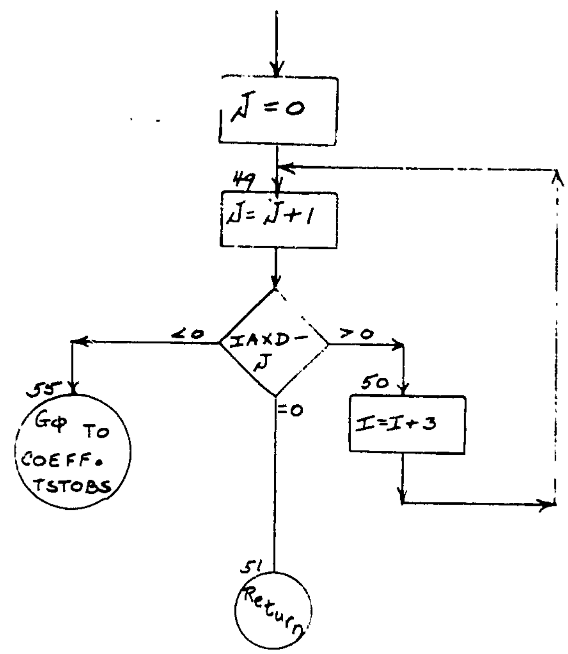


SUBROUTINE: REAL

Read Station Cards from TAPE 10

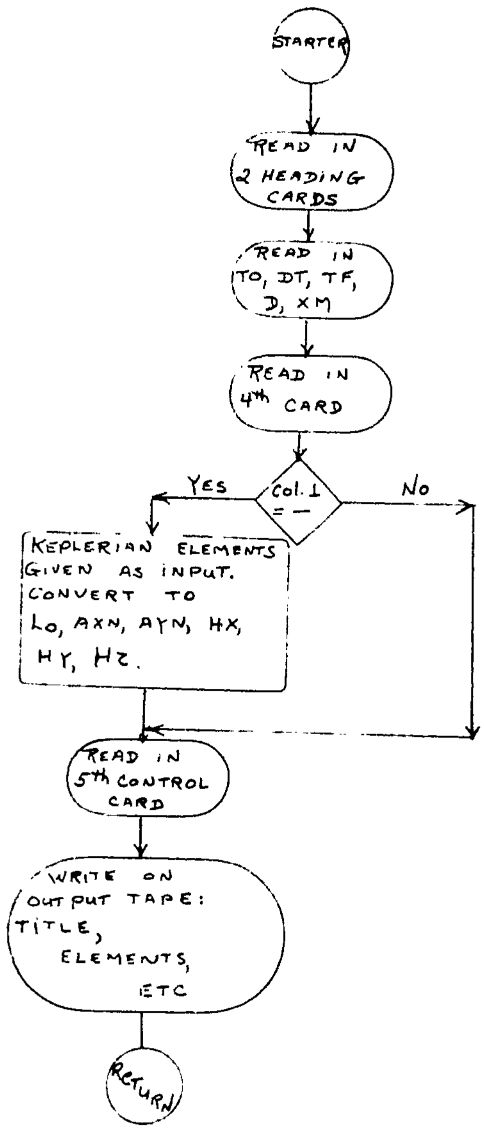


Subroutine SETI

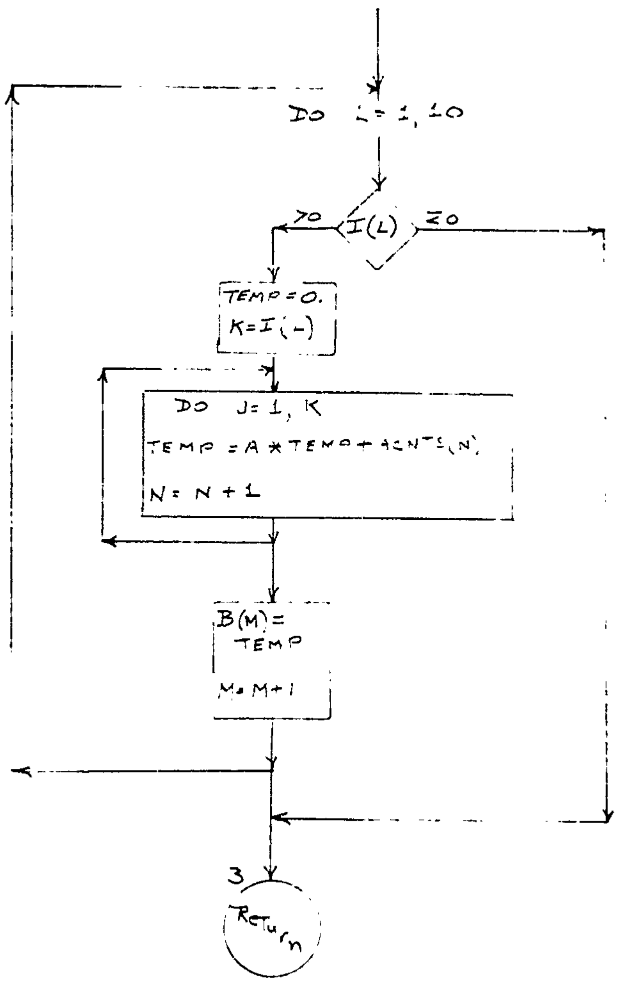


SUBROUTINE. STARTER

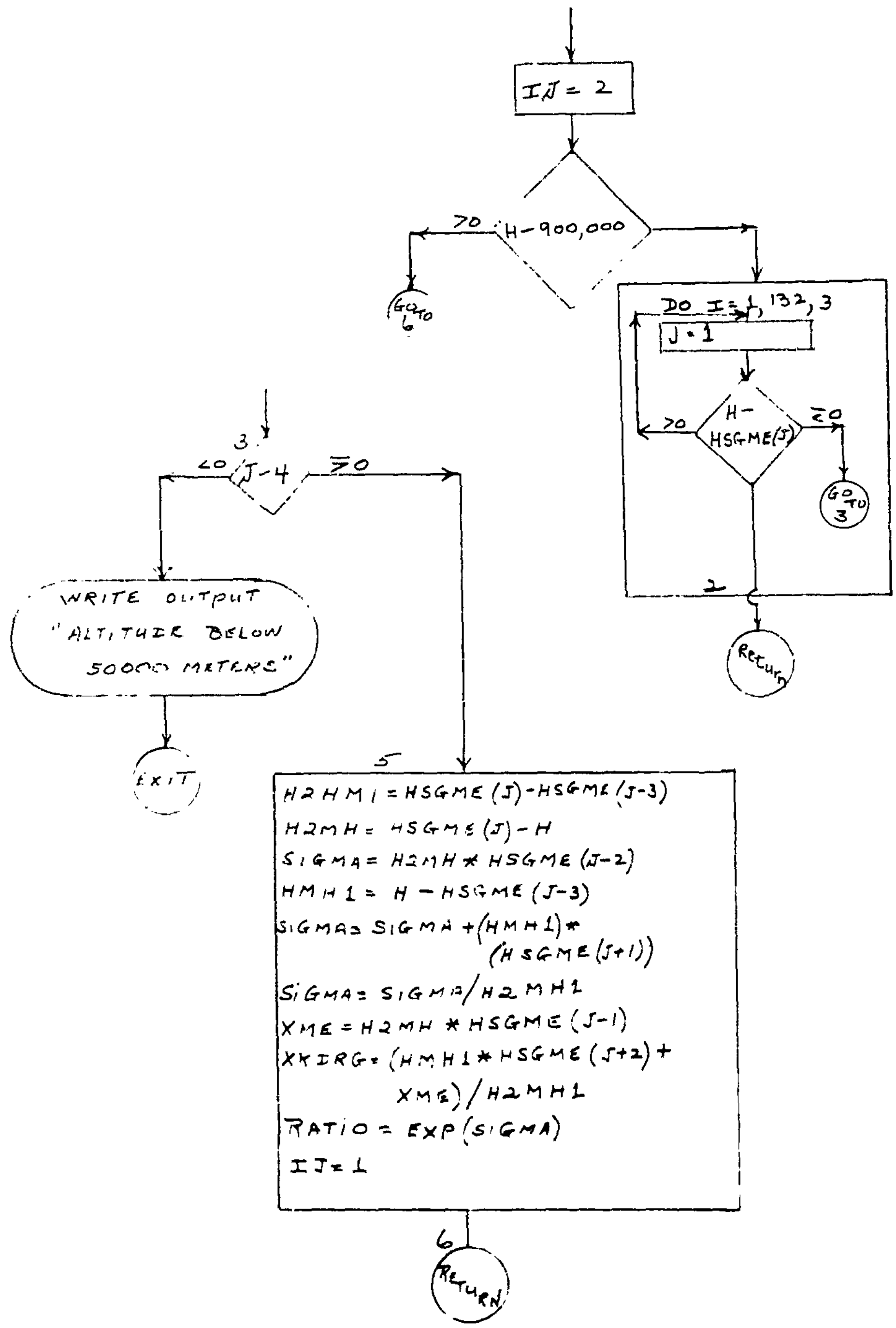
READ 1ST FIVE CONTROL CARDS



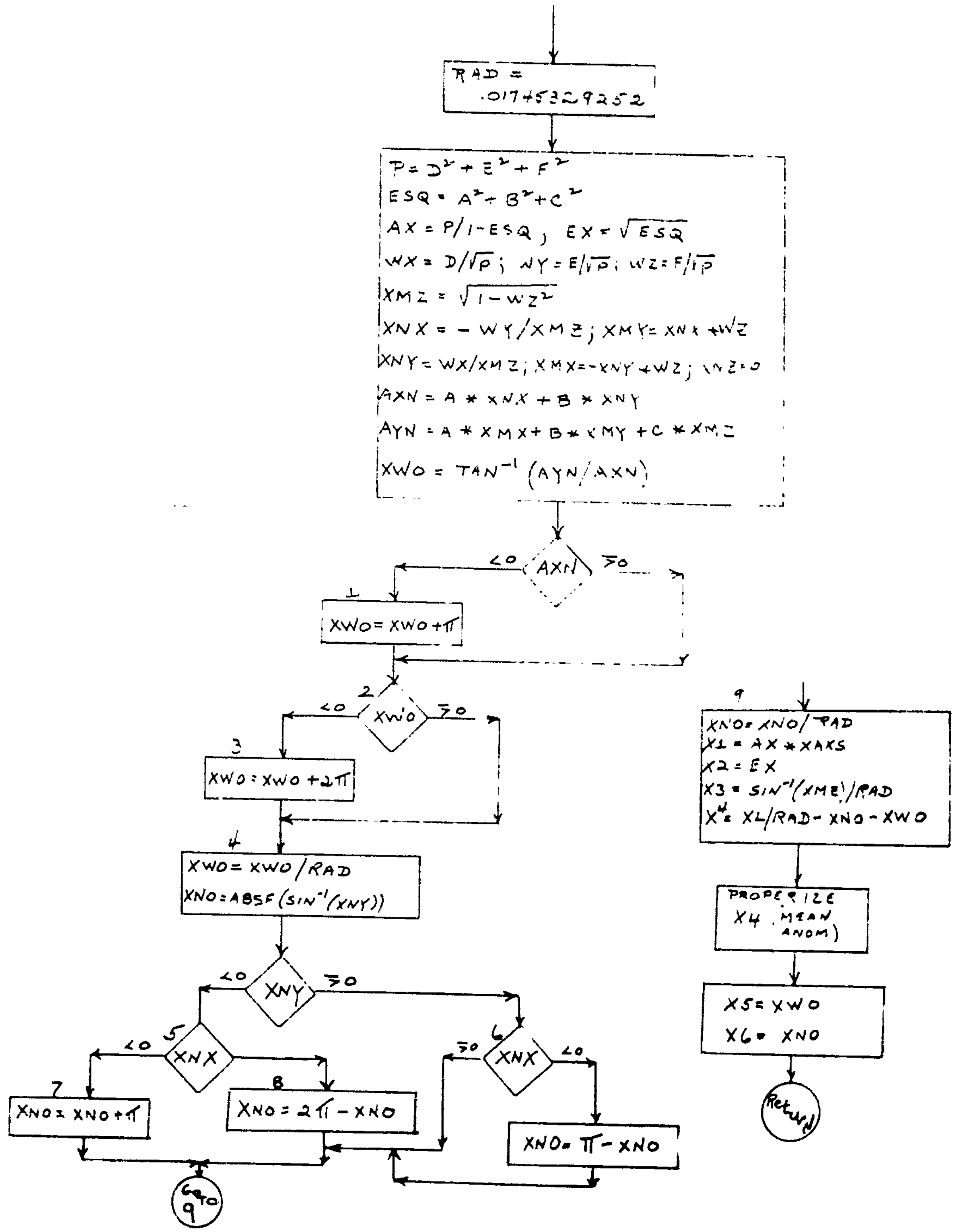
Subroutine SU



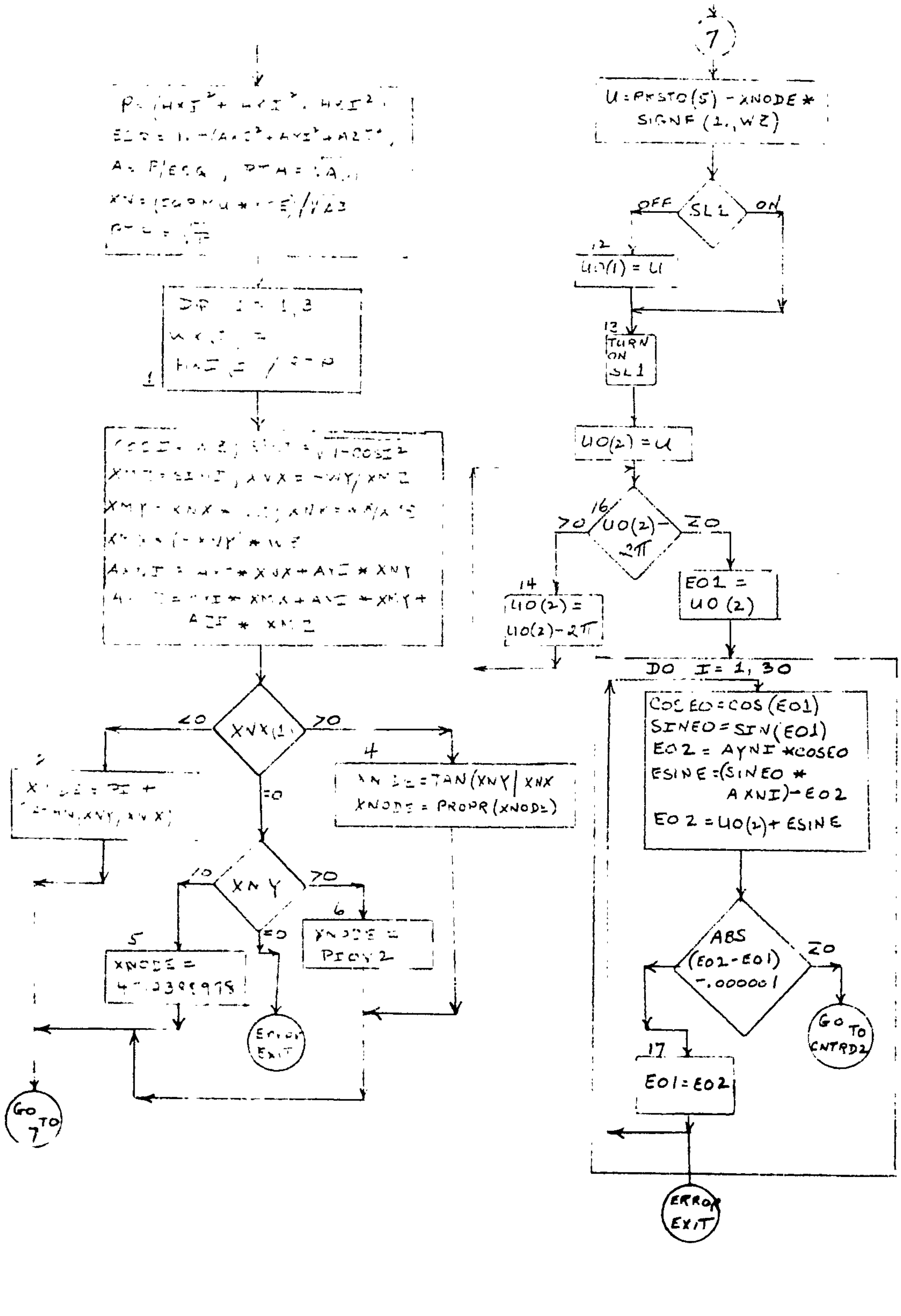
Subroutine TBINT



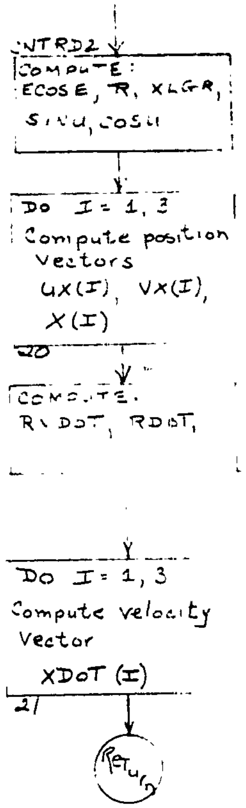
Subroutine XEL



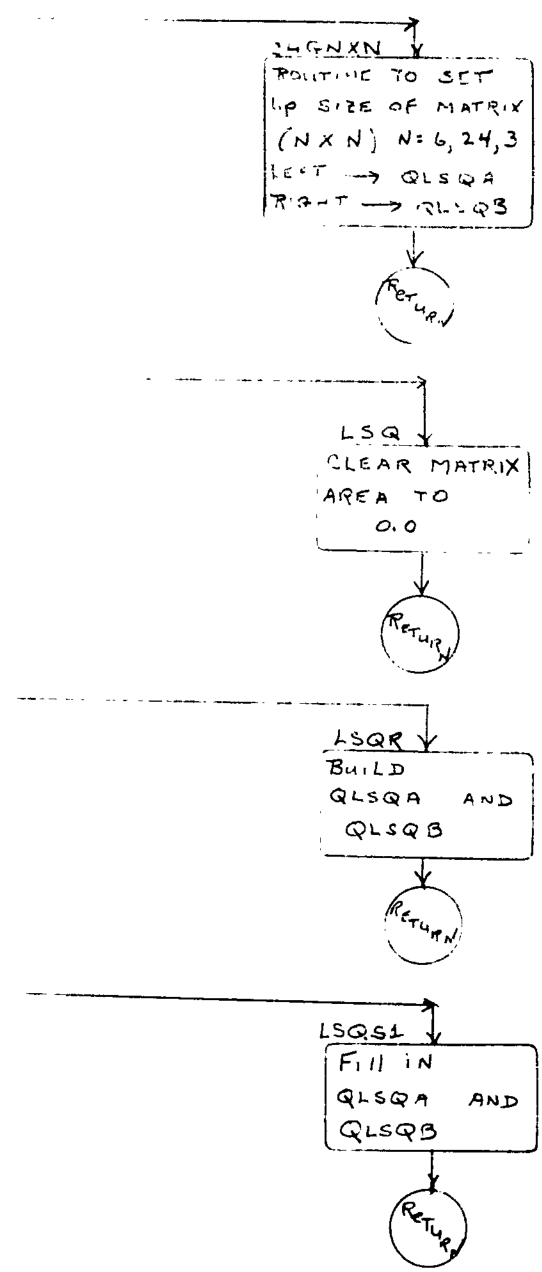
Subroutine XYZ...B

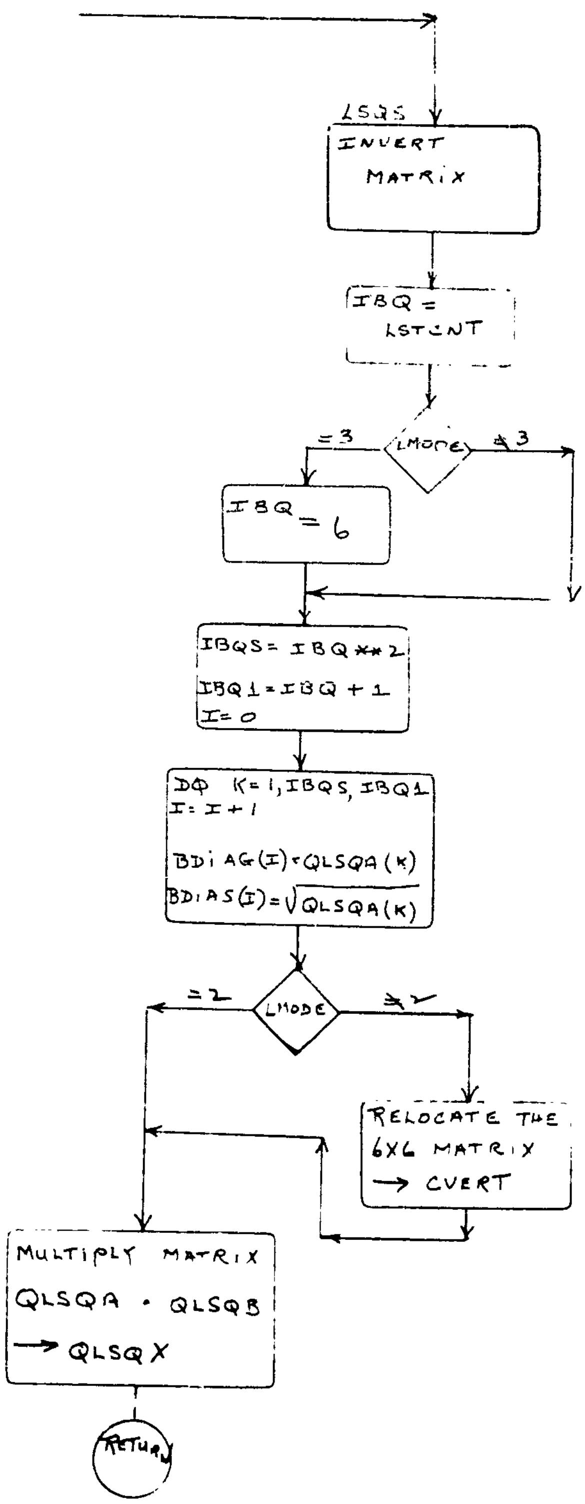


XYZSB (CONT)

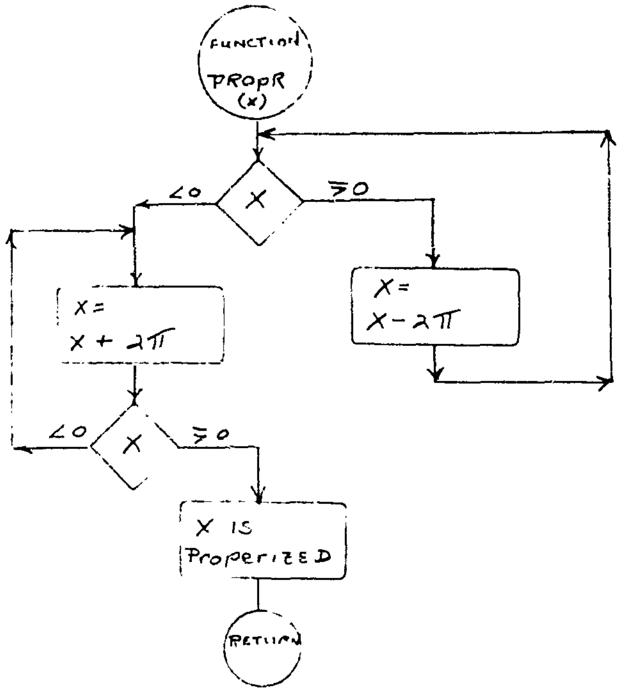

INTRD2
COMPUTE:
ECOSE, R, XLGR,
SINU, COSU
DO I = 1, 3
Compute position
Vectors
UX(I), VX(I),
X(I)
20
COMPUTE:
RIDOT, RDOT,
DO I = 1, 3
Compute velocity
Vector
XDOT (I)
21
Return

LEAST SQUARES SUBROUTINE

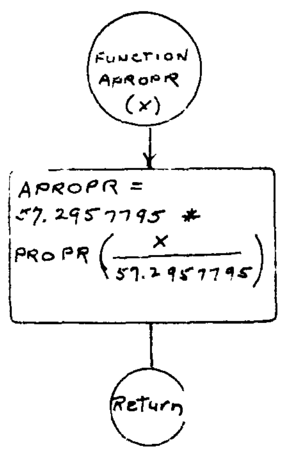



LSQS
INVERT MATRIX
IBQ = LSTCNT
LMODE
=3
≠3
IBQ = 6
IBQS = IBQ**2
IBQ1 = IBQ + 1
I = 0
DΦ K = 1, IBQS, IBQ1
I = I + 1
BDIAG(I) = QLSQA(K)
BDIAS(I) = √QLSQA(K)
LMODE
=2
≠2
RELOCATE THE 6X6 MATRIX → CVERT
MULTIPLY MATRIX
QLSQA · QLSQB
→ QLSQX
RETURN

PROPERIZES ANGLE GIVEN IN RADIANS

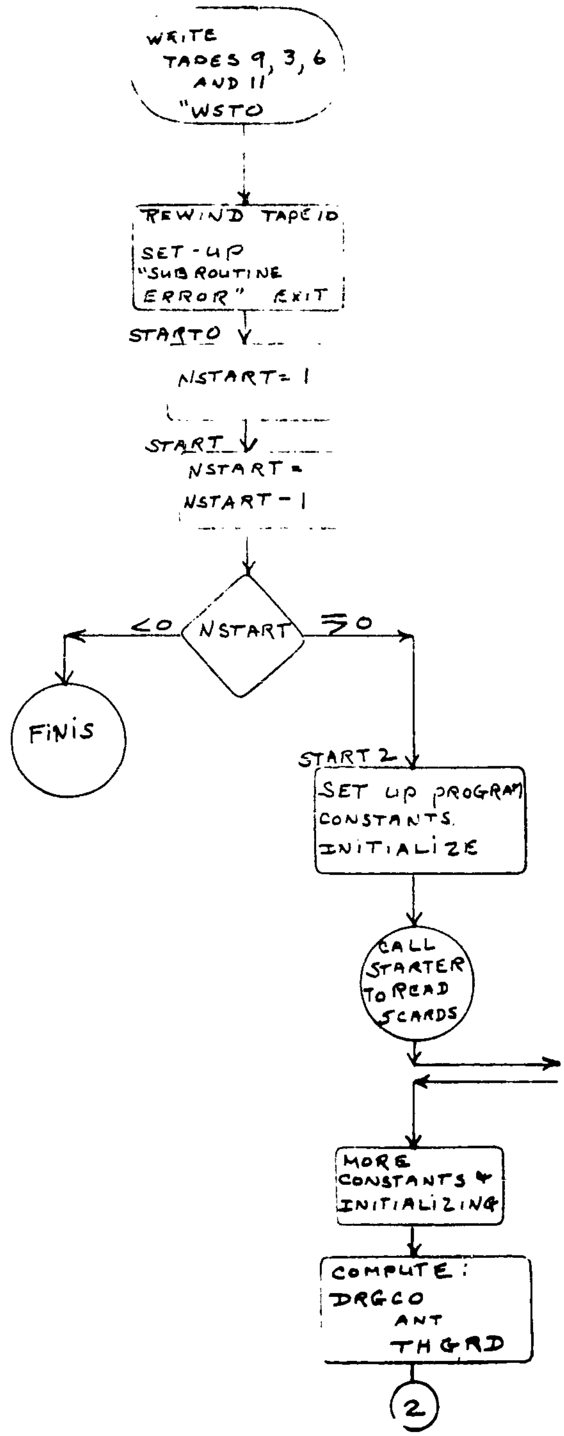


PROPERIZES ANGLE GIVEN IN DEGREES

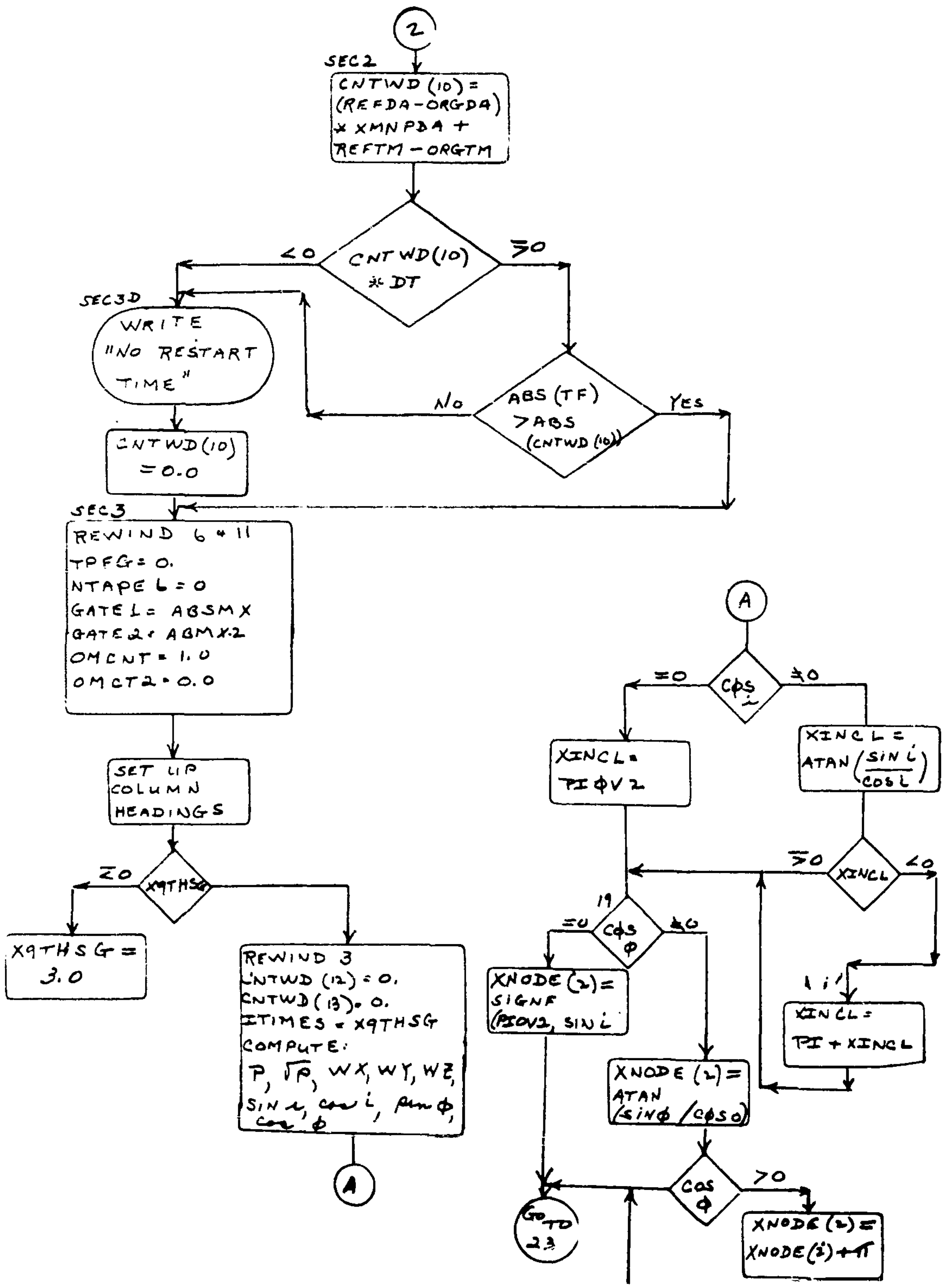


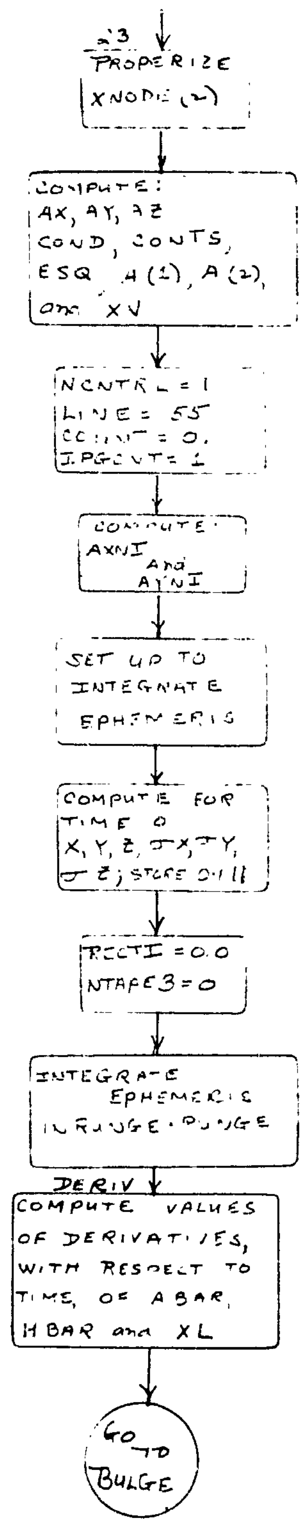
IX

FLOW CHART OF MAIN PROGRAM

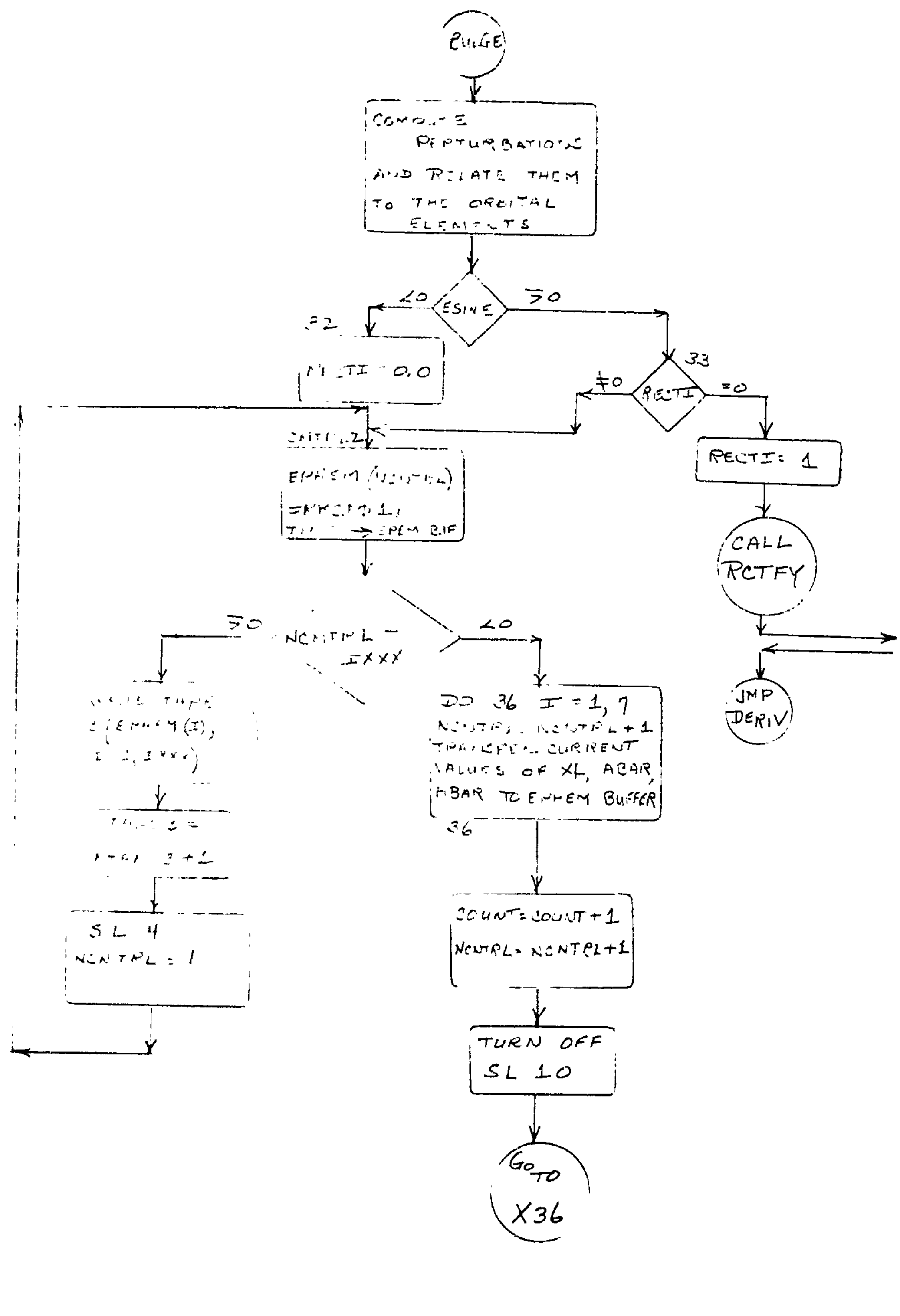


WRITE TAPES 9, 3, 6 AND 11 "WSTO
REWIND TAPE 10
SET-UP "SUBROUTINE ERROR" EXIT
STARTO
NSTART= 1
START
NSTART = NSTART - 1
<0
NSTART
≥0
FINIS
START 2
SET UP PROGRAM CONSTANTS. INITIALIZE
CALL STARTER TO READ 5 CARDS
MORE CONSTANTS & INITIALIZING
COMPUTE: DRGCO AND THGRD
2

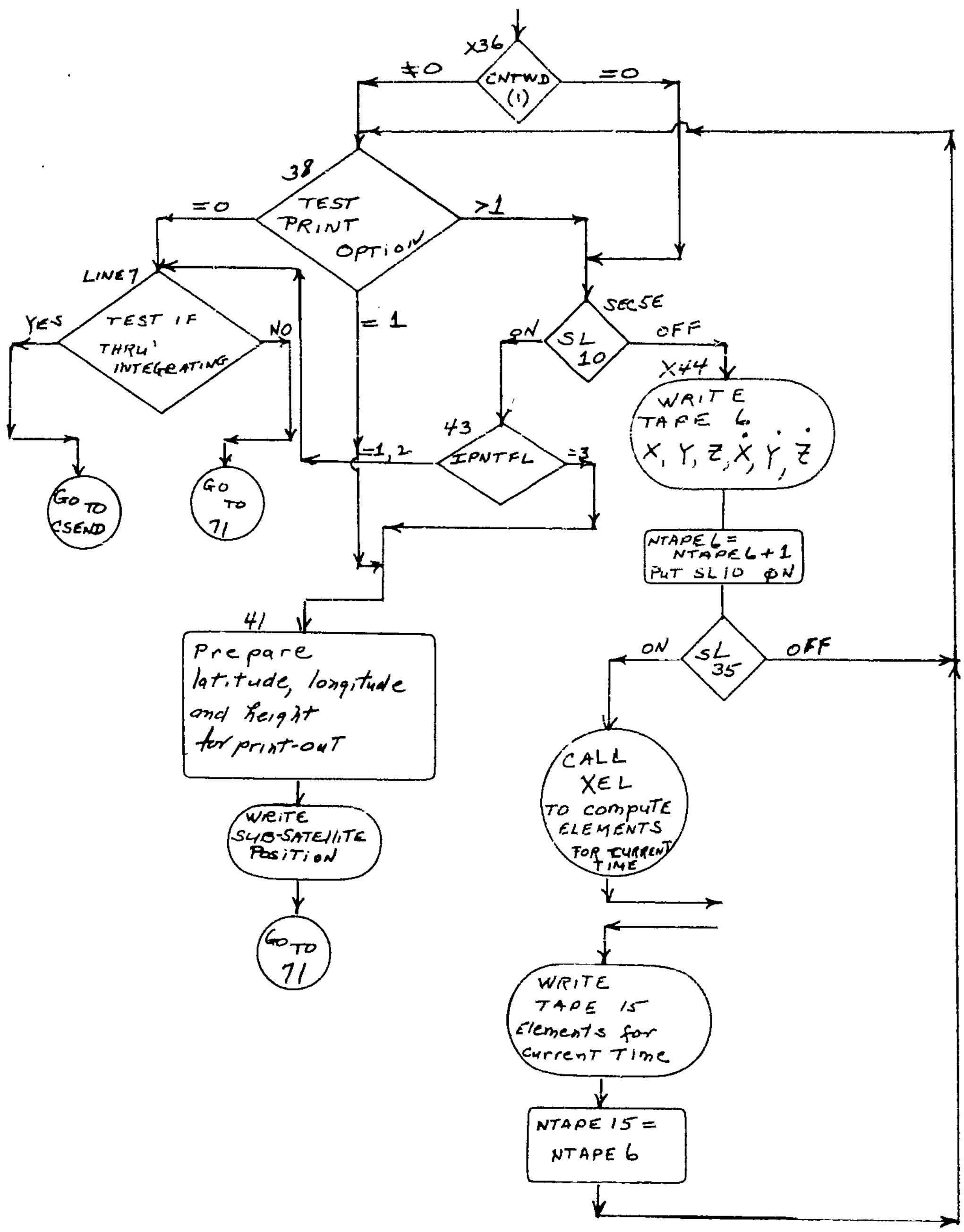

2
SEC2
CNTWD (10) = (REFDA-ORGDA) * XMNPDA + REFTM - ORGTM
CNTWD(10) * DT
<0
≥0
SEC3D
WRITE "NO RESTART TIME"
No
ABS (TF) > ABS (CNTWD (10))
YES
CNTWD(10) = 0.0
SEC3
REWIND 6 + 11
TPFG = 0.
NTAPE 6 = 0
GATE 1 = ABSMX
GATE 2 = ABMX2
OMCNT = 1.0
OMCT2 = 0.0
SET UP COLUMN HEADINGS
≥0
X9THSG
X9THSG = 3.0
REWIND 3
CNTWD (12) = 0.
CNTWD (13) = 0.
ITIMES = X9THSG
COMPUTE:
P, √P, WX, WY, WZ, SIN i, cos i, sin φ, cos φ
A
A
=0
COS i
≠0
XINCL = PIOV2
XINCL = ATAN (SIN i / COS i)
≥0
XINCL
<0
19
=0
COS φ
≠0
XNODE (2) = SIGNF (PIOV2, SIN i)
XINCL = PI + XINCL
XNODE (2) = ATAN (SIN φ / COS φ)
COS φ
>0
GO TO 23
XNODE (2) = XNODE (2) + π

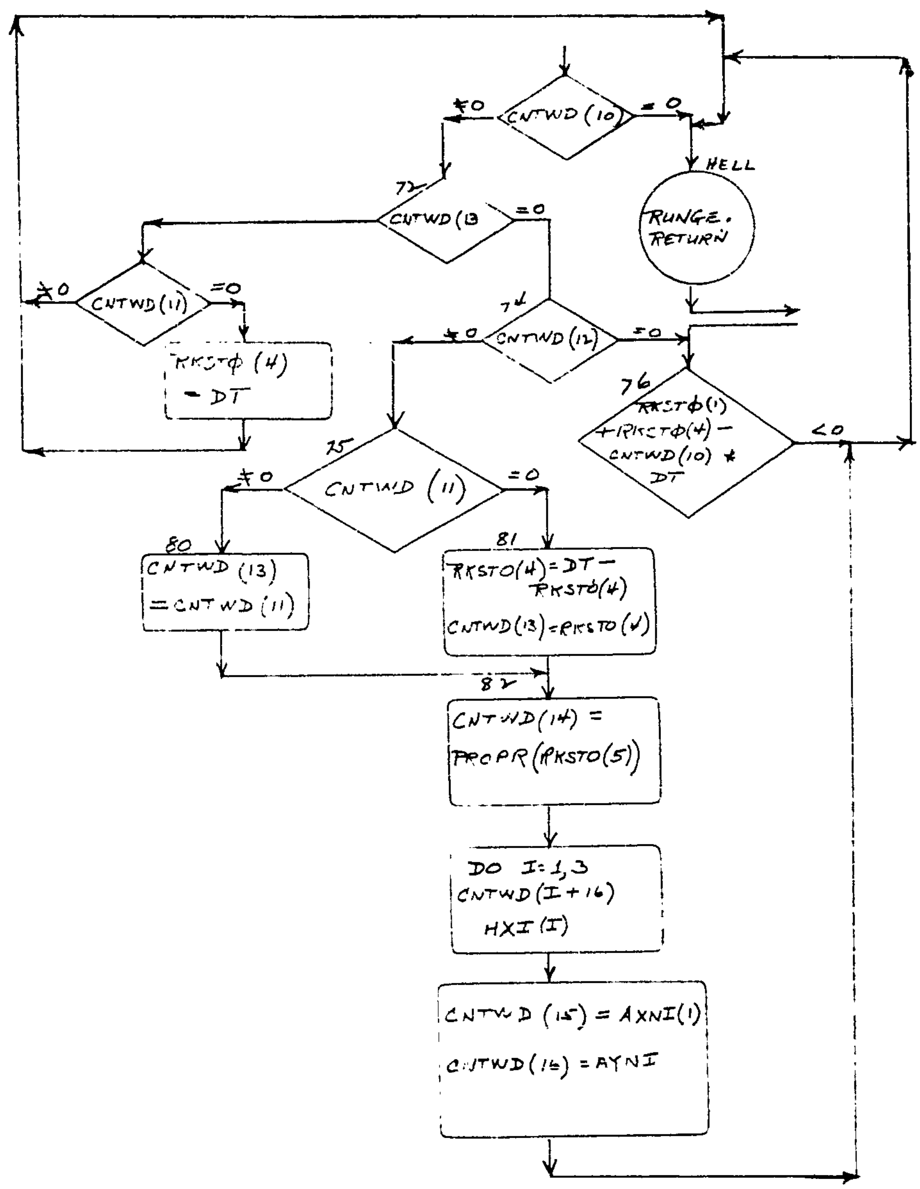


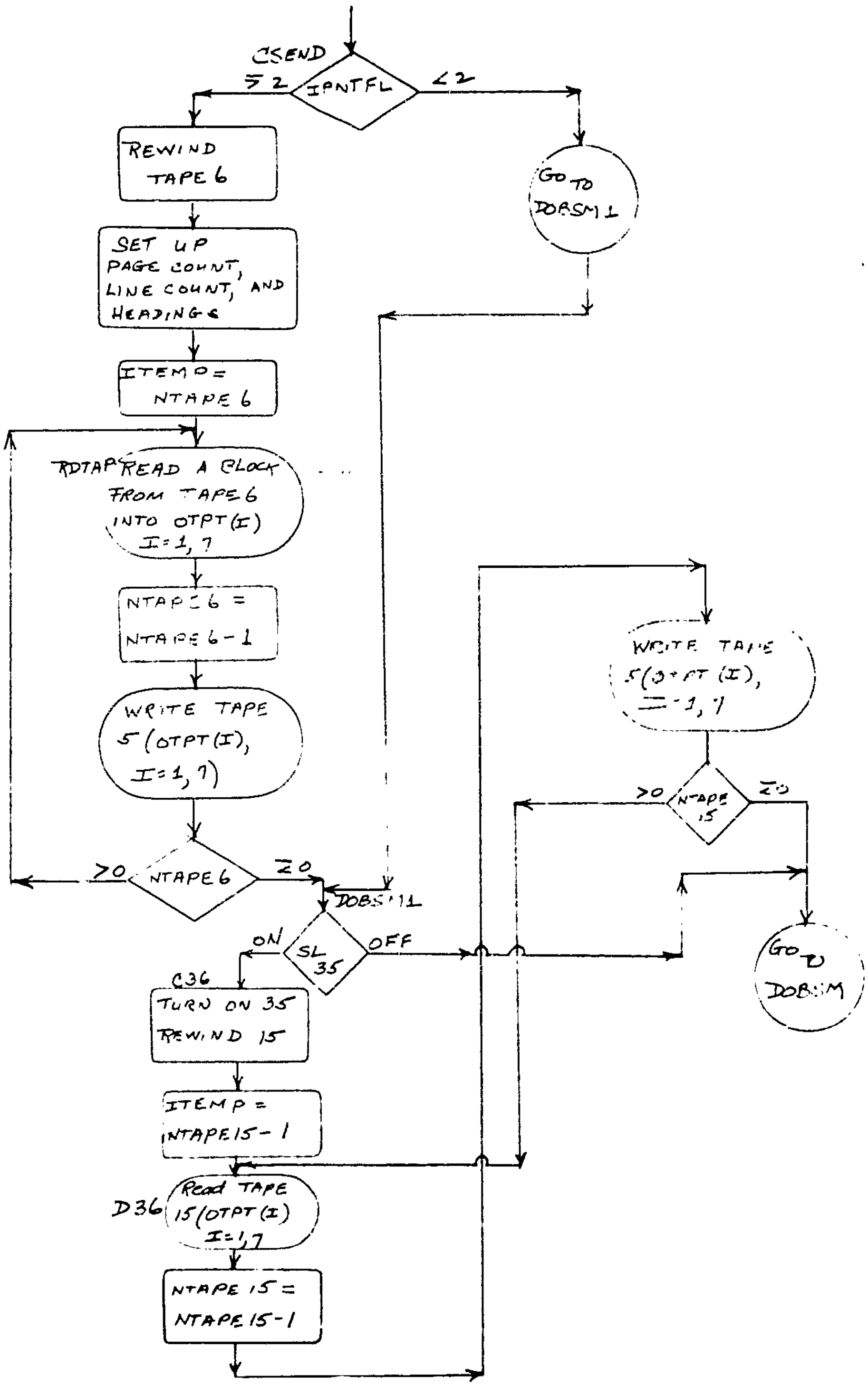
23
PROPERIZE
XNODE (2)
COMPUTE:
AX, AY, AZ
COND, CONTS,
ESQ, A(1), A(2),
and XV
NCNTRL = 1
LINE = 55
CCOUNT = 0.
IPGCNT = 1
COMPUTE:
AXNI and AYNI
SET UP TO
INTEGRATE
EPHEMERIS
COMPUTE FOR
TIME 0
X, Y, Z, JX, JY,
JZ; STORE 0-11
RECTI = 0.0
NTAPE3 = 0
INTEGRATE
EPHEMERIS
IN RUNGE-KUNGE
DERIV
COMPUTE VALUES
OF DERIVATIVES,
WITH RESPECT TO
TIME, OF ABAR,
HBAR and XL
GO TO
BULGE

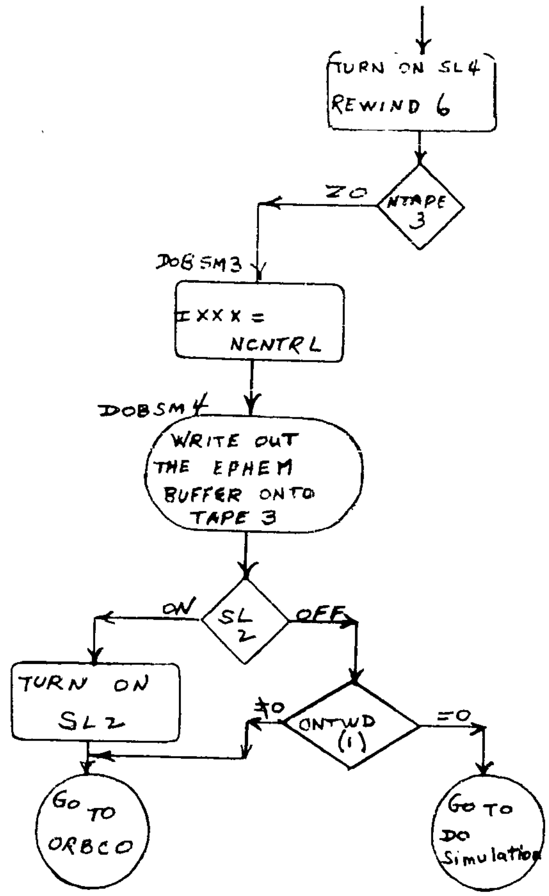


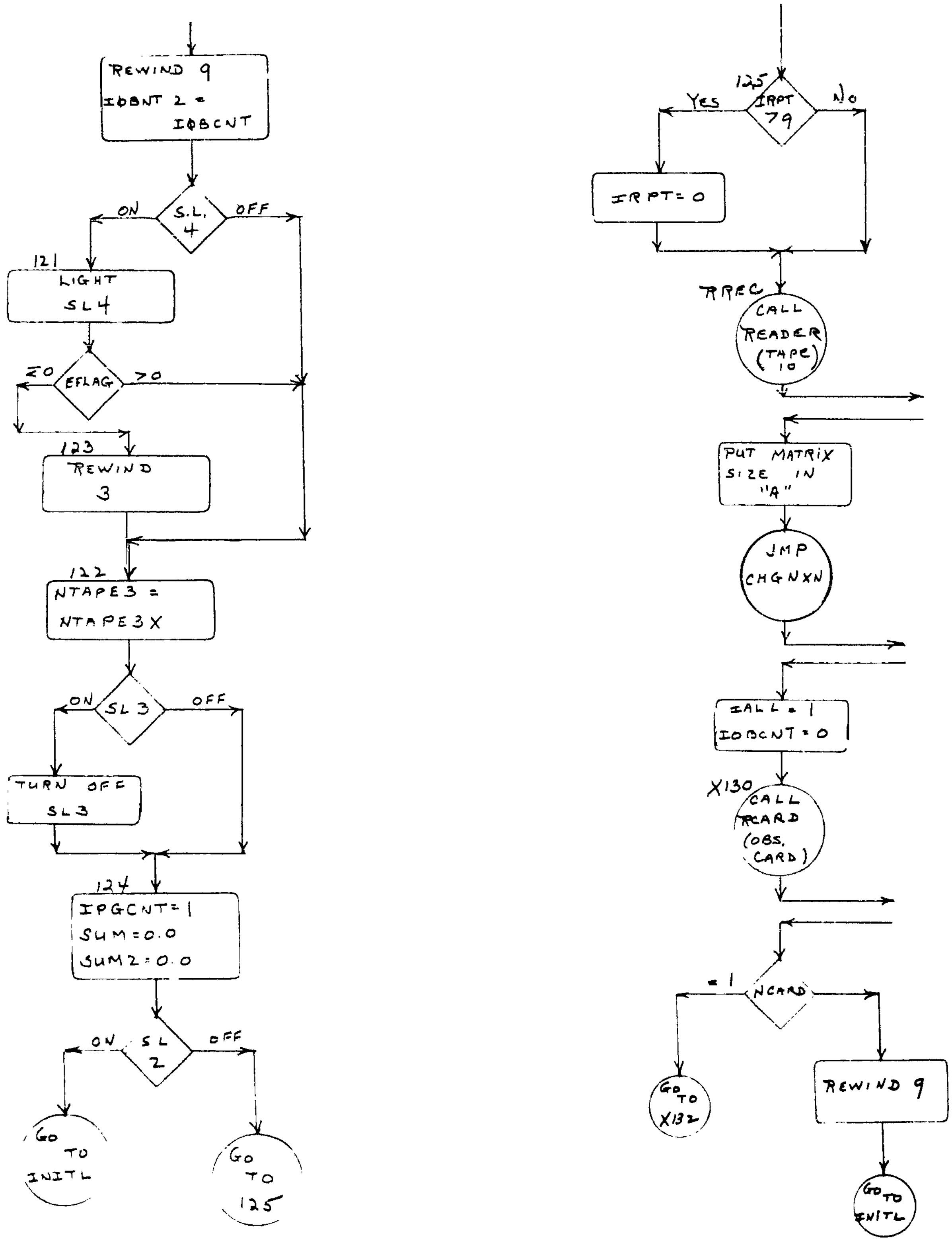
BULGE
COMPUTE PERTURBATIONS AND RELATE THEM TO THE ORBITAL ELEMENTS
ESINE
RECTI
RECTI = 1
CALL RCTFY
JMP DERIV
DO 36 I = 1, 7
36
COUNT = COUNT + 1
NCNTRL = NCNTRL + 1
TURN OFF SL 10
GO TO X36
SL 4


X36
CNTWD (1)
≠0
=0
38
TEST PRINT OPTION
=0
>1
=1
LINE 7
TEST IF THRU' INTEGRATING
YES
NO
GO TO CSEND
GO TO 71
SECSE
SL 10
ON
OFF
X44
WRITE TAPE 6 X, Y, Z, Ẋ, Ẏ, Ż
43
IPNTFL
=1, 2
=3
NTAPE 6 = NTAPE 6 + 1 PUT SL 10 ON
SL 35
ON
OFF
41
Prepare latitude, longitude and height for print-out
WRITE SUB-SATELLITE POSITION
GO TO 71
CALL XEL TO COMPUTE ELEMENTS FOR CURRENT TIME
WRITE TAPE 15 Elements for Current Time
NTAPE 15 = NTAPE 6

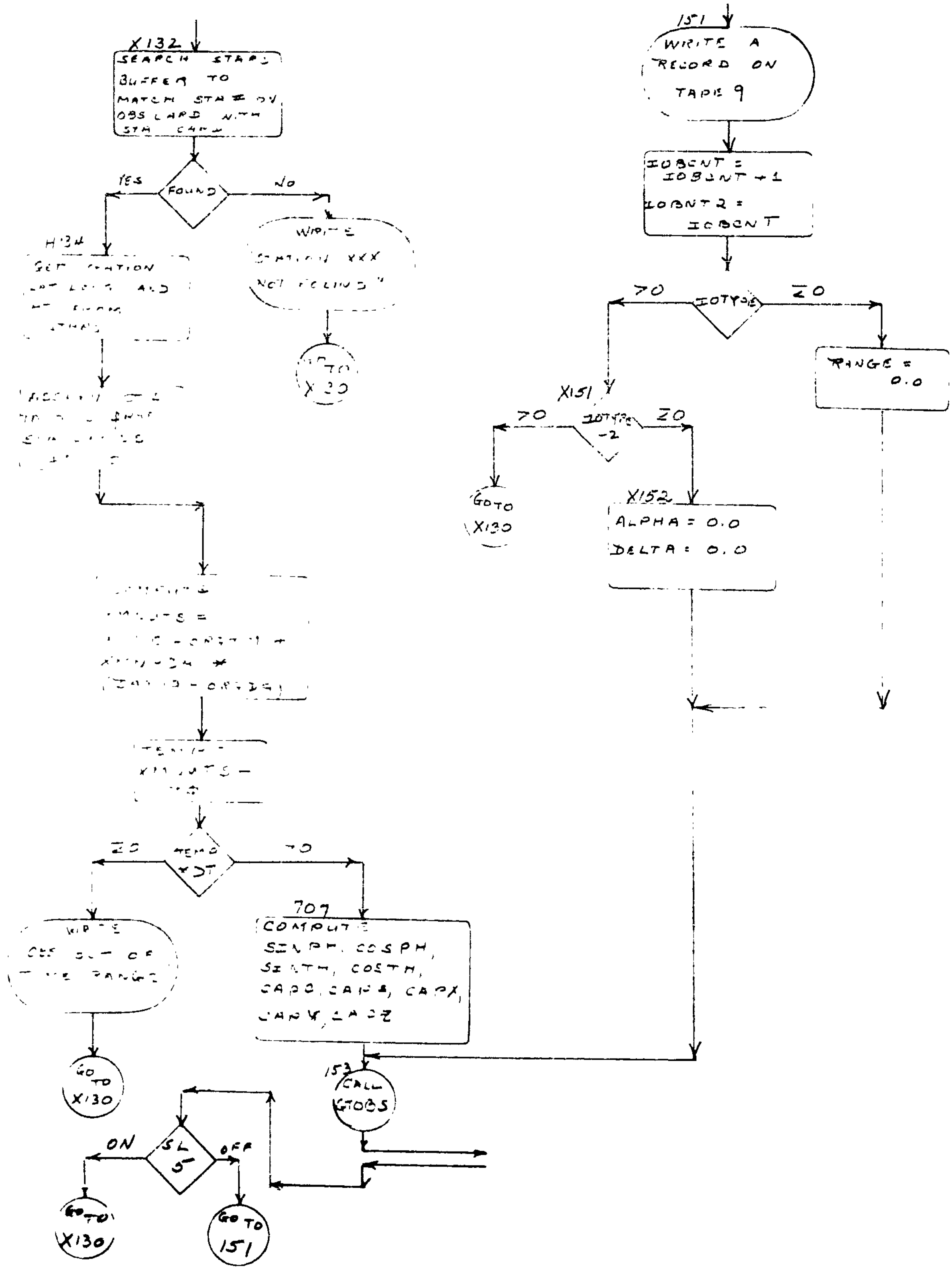

CNTWD (10)
≠0
=0
HELL
RUNGE. RETURN
72
CNTWD (13
=0
CNTWD (11)
≠0
=0
RKSTΦ (4)
= DT
74
CNTWD (12)
≠0
=0
76
RKSTΦ(1)
+RKSTΦ(4) −
CNTWD(10) *
DT
<0
75
CNTWD (11)
≠0
=0
80
CNTWD (13)
= CNTWD (11)
81
RKSTO(4) = DT −
RKSTO(4)
CNTWD(13) = RKSTO (4)
82
CNTWD(14) =
PROPR (RKSTO(5))
DO I=1,3
CNTWD(I+16)
HXI (I)
CNTWD (15) = AXNI(1)
CNTWD (16) = AYNI

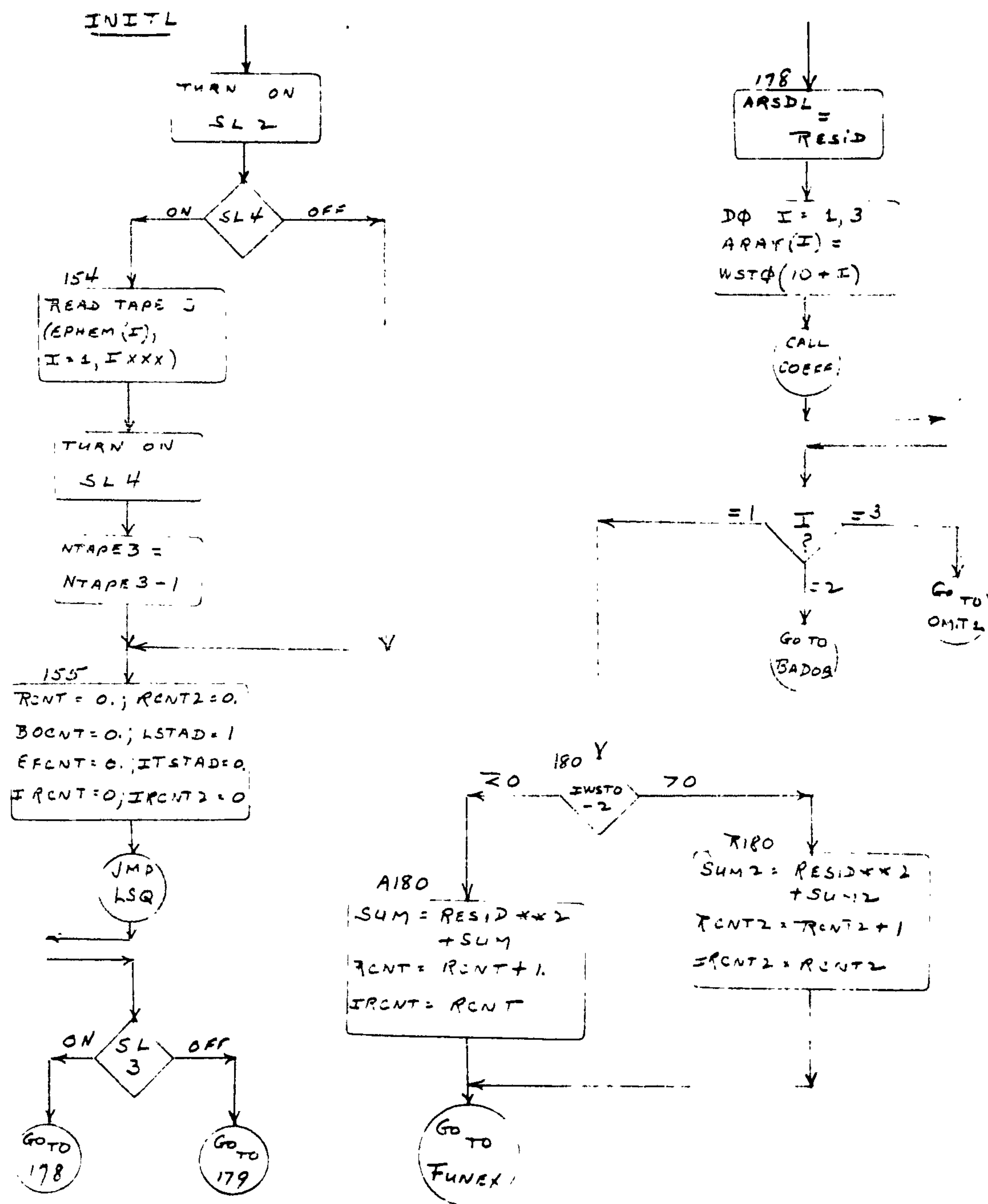

CSEND
IPNTFL
≥ 2
< 2
REWIND TAPE 6
GO TO DOBSM1
SET UP PAGE COUNT, LINE COUNT, AND HEADINGS
ITEMP = NTAPE 6
RDTAP READ A BLOCK FROM TAPE 6 INTO OTPT(I) I=1, 7
NTAPE 6 = NTAPE 6-1
WRITE TAPE 5 (OTPT(I), I=1, 7)
NTAPE 6
>0
≤0
DOBSM1
SL 35
ON
OFF
C36
TURN ON 35 REWIND 15
ITEMP = NTAPE15-1
D36
Read TAPE 15 (OTPT(I) I=1,7
NTAPE 15 = NTAPE 15-1
NTAPE 15
GO TO DOBSM

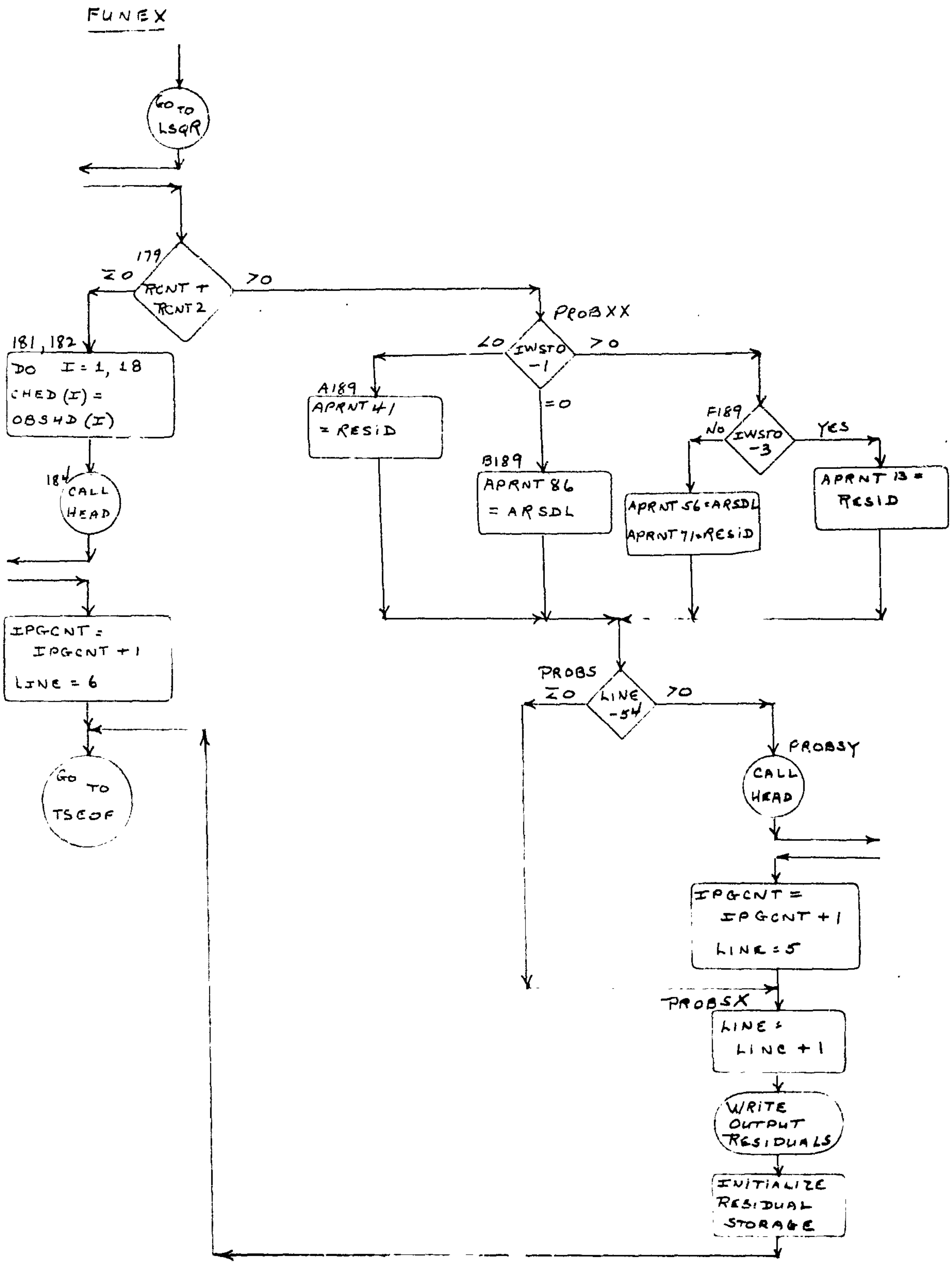

TURN ON SL4
REWIND 6
≠0
NTAPE 3
DOBSM3
IXXX = NCNTRL
DOBSM4
WRITE OUT THE EPHEM BUFFER ONTO TAPE 3
ON
SL 2
OFF
TURN ON SL2
≠0
ONTWD (1)
=0
Go TO ORBCO
Go TO DO Simulation

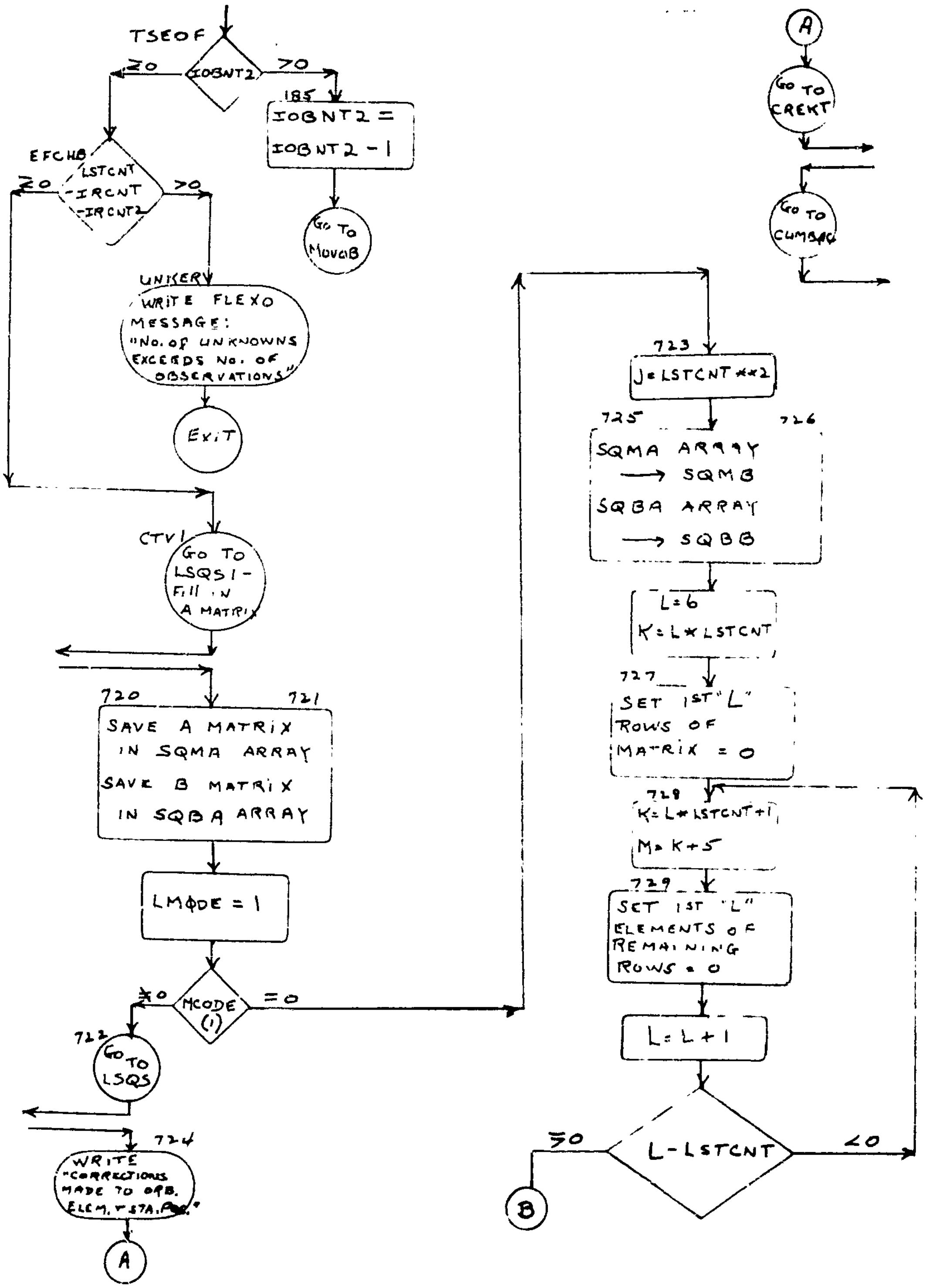


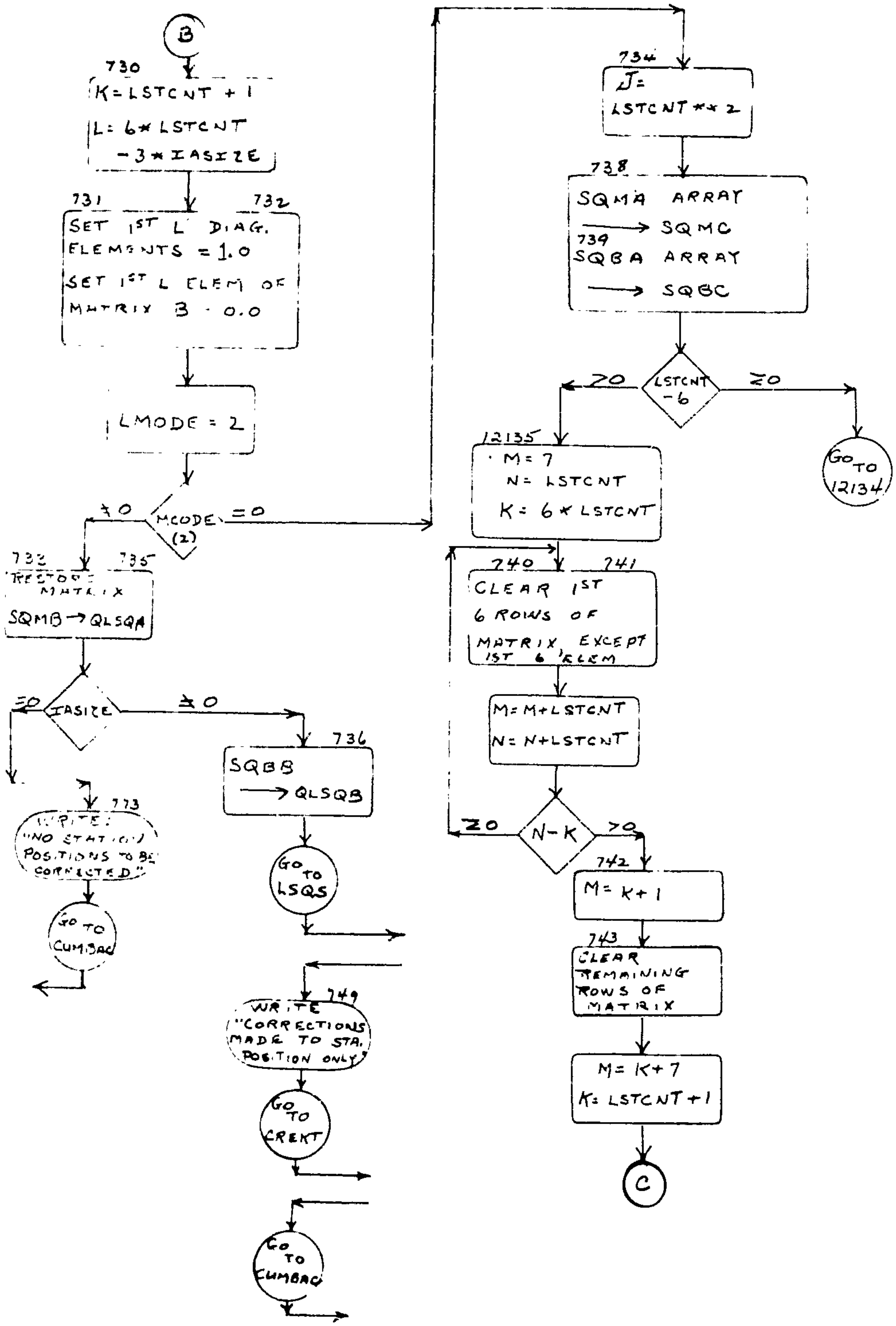
REWIND 9
IØBNT 2 = IØBCNT
ON
S.L. 4
OFF
121
LIGHT SL4
≤0
EFLAG
>0
123
REWIND 3
122
NTAPE3 = NTAPE3X
ON
SL 3
OFF
TURN OFF SL3
124
IPGCNT=1
SUM=0.0
SUM2=0.0
ON
SL 2
OFF
GO TO INITL
GO TO 125
125
Yes
IRPT 79
No
IRPT=0
RREC
CALL READER (TAPE 10)
PUT MATRIX SIZE IN "A"
JMP CHGNXN
IALL = 1
IOBCNT = 0
X130
CALL RCARD (OBS. CARD)
= 1
NCARD
GO TO X132
REWIND 9
GO TO INITL

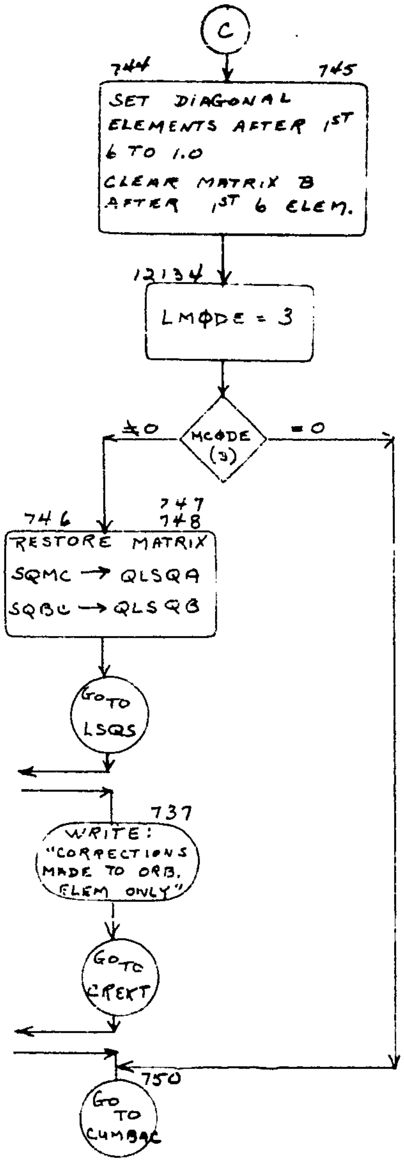

X132
SEARCH STARS BUFFER TO MATCH STA # ON OBS CARD WITH STA CARD
FOUND
YES
NO
H134
WRITE
GO TO X130
TEMP
+DT
707
COMPUTE SINPH, COSPH, SINTH, COSTH,
GO TO X130
153
CALL GTOBS
SL 5
ON
OFF
GO TO X130
GO TO 151
151
WRITE A RECORD ON TAPE 9
IOBCNT = IOBCNT + 1
IOBNT2 = IOBCNT
IOTYPE
RANGE = 0.0
X151
IOTYPE -2
GO TO X130
X152
ALPHA = 0.0
DELTA = 0.0

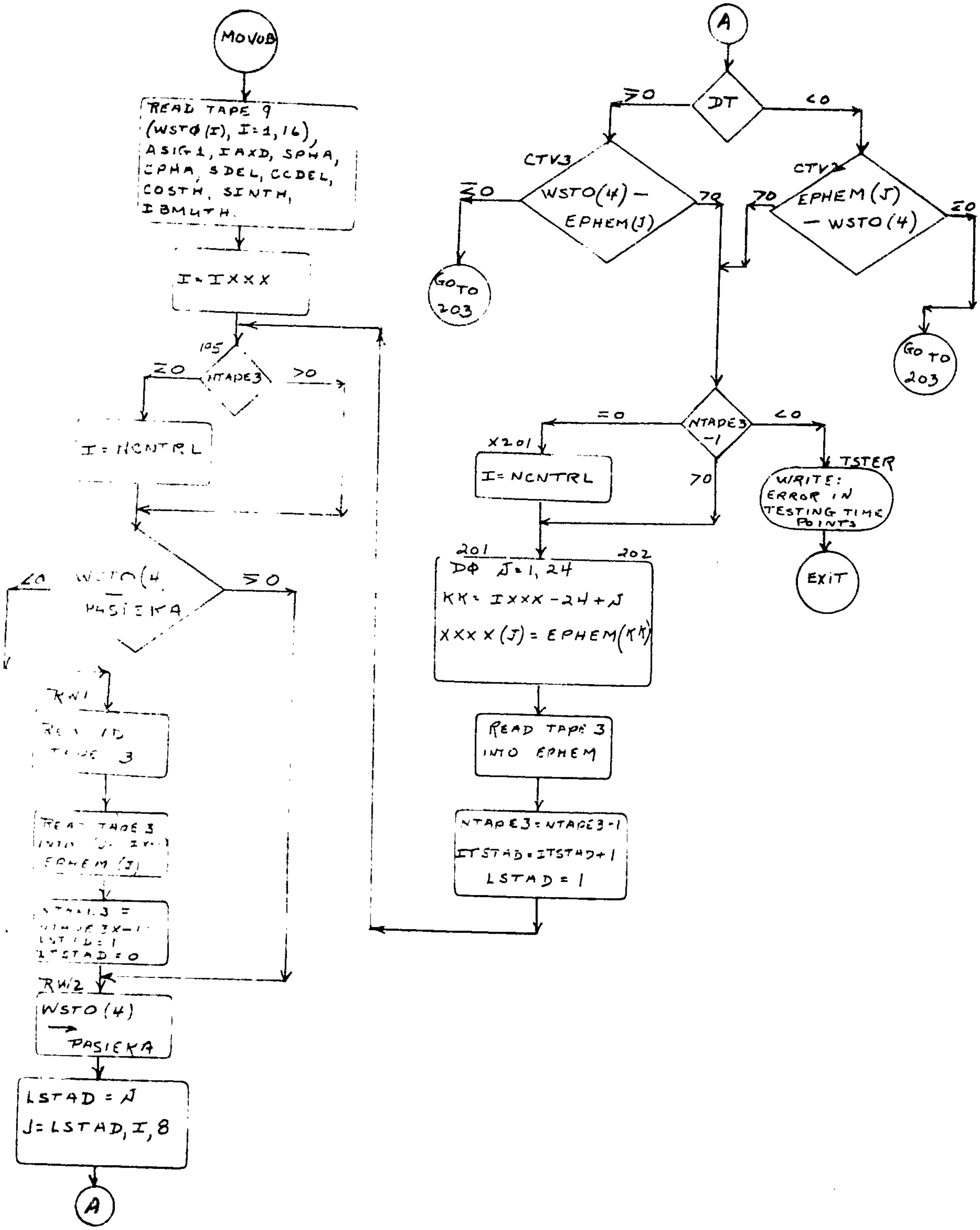

INITL
TURN ON SL 2
SL 4
ON
OFF
154
READ TAPE 3 (EPHEM (I), I = 1, IXXX)
TURN ON SL 4
NTAPE3 = NTAPE 3 - 1
155
RCNT = 0.; RCNT2 = 0.
BOCNT = 0.; LSTAD = 1
EFCNT = 0.; ITSTAD = 0
IRCNT = 0; IRCNT2 = 0
JMP LSQ
SL 3
ON
OFF
GO TO 178
GO TO 179
178
ARSDL = RESID
DØ I = 1, 3
ARAY(I) = WSTØ(10 + I)
CALL COEFF
I ?
= 1
= 2
= 3
GO TO BADOB
GO TO OMIT2
180
IWSTO -2
≤ 0
> 0
A180
SUM = RESID ** 2 + SUM
RCNT = RCNT + 1.
IRCNT = RCNT
R180
SUM2 = RESID ** 2 + SUM2
RCNT2 = RCNT2 + 1
IRCNT2 = RCNT2
GO TO FUNEX


FUNEX
GO TO LSQR
179
RCNT + RCNT 2
≥0
>0
181, 182
DO I = 1, 18
CHED (I) = OBSHD (I)
184
CALL HEAD
IPGCNT = IPGCNT + 1
LINE = 6
GO TO TSCOF
PROBXX
IWSTO -1
<0
>0
=0
A189
APRNT 41 = RESID
B189
APRNT 86 = ARSDL
F189
IWSTO -3
NO
YES
APRNT 56 = ARSDL
APRNT 71 = RESID
APRNT 13 = RESID
PROBS
LINE -54
≥0
>0
PROBSY
CALL HEAD
IPGCNT = IPGCNT + 1
LINE = 5
PROBSX
LINE = LINE + 1
WRITE OUTPUT RESIDUALS
INITIALIZE RESIDUAL STORAGE

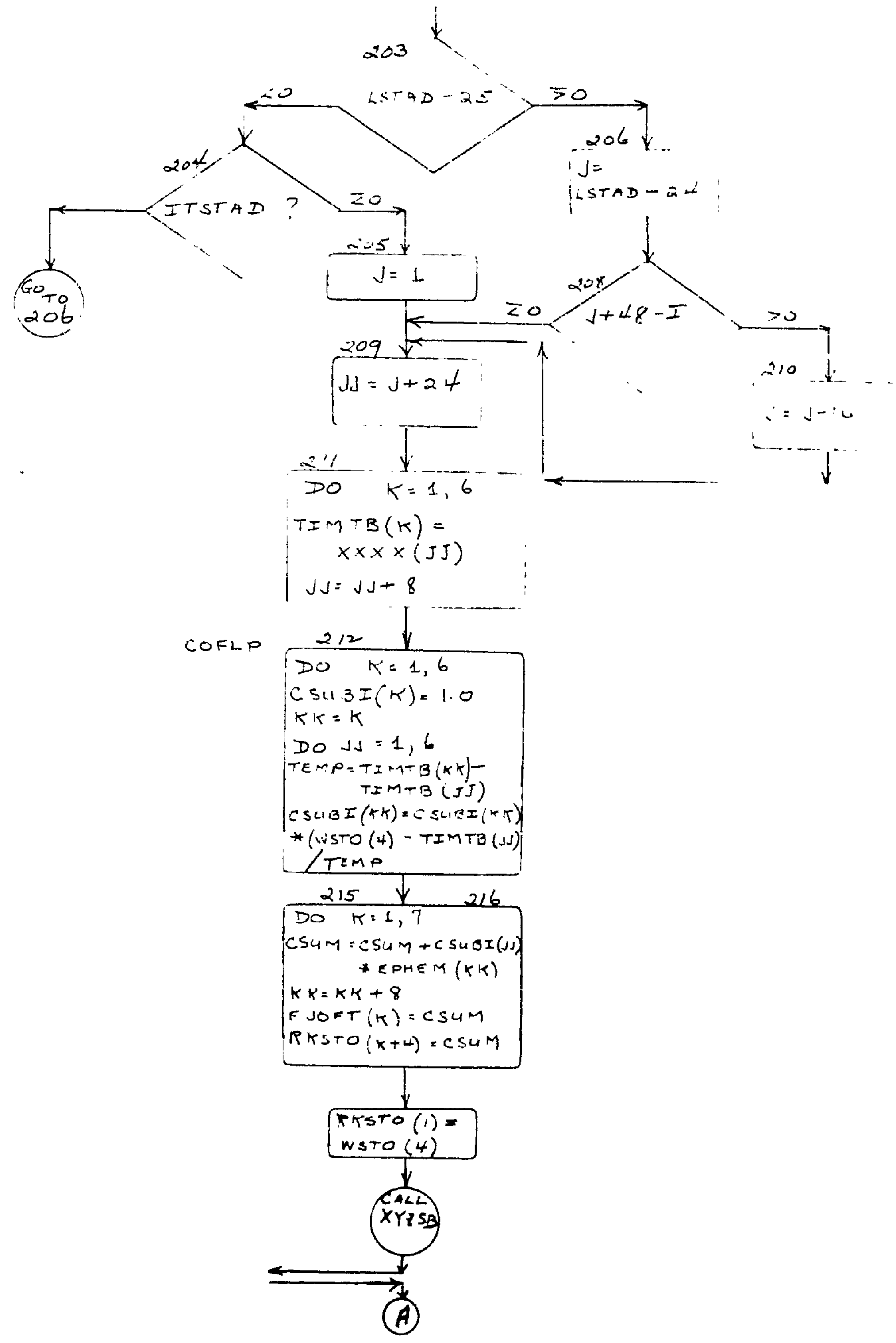

TSEOF
IOBNT2
≥0
>0
185
IOBNT2 = IOBNT2 - 1
Go To MOVAB
EFCHB
LSTCNT -IRCNT -IRCNT2
≤0
>0
UNKERR
WRITE FLEXO MESSAGE: "No. of UNKNOWNS EXCEEDS No. OF OBSERVATIONS"
EXIT
CTV1
Go To LSQS1 - Fill IN A MATRIX
720
721
SAVE A MATRIX IN SQMA ARRAY
SAVE B MATRIX IN SQBA ARRAY
LMODE = 1
MCODE (1)
≠0
=0
722
Go To LSQS
724
WRITE "CORRECTIONS MADE TO ORB. ELEM. + STA. POS."
A
Go To CREKT
Go To CUMBAC
723
J = LSTCNT **2
725
726
SQMA ARRAY → SQMB
SQBA ARRAY → SQBB
L=6
K=L*LSTCNT
727
SET 1ST "L" ROWS OF MATRIX = 0
728
K=L*LSTCNT+1
M=K+5
729
SET 1ST "L" ELEMENTS OF REMAINING ROWS = 0
L=L+1
L-LSTCNT
≥0
<0
B

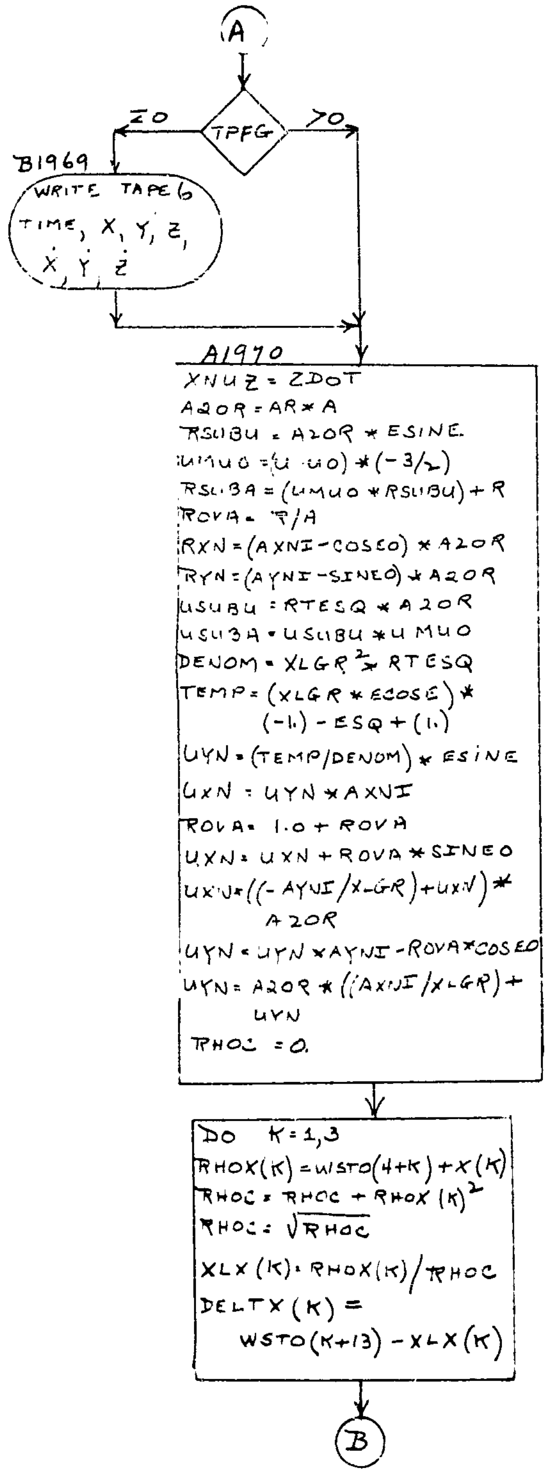

B
730
K = LSTCNT + 1
L: 6 * LSTCNT - 3 * IASIZE
731
732
SET 1ST L DIAG. ELEMENTS = 1.0
SET 1ST L ELEM OF MATRIX B = 0.0
LMODE = 2
MCODE (2)
≠0
=0
733
735
RESTORE MATRIX
SQMB → QLSQA
IASIZE
=0
≠0
736
SQBB → QLSQB
GO TO LSQS
773
WRITE "NO STATION POSITIONS TO BE CORRECTED"
GO TO CUMBAC
749
WRITE "CORRECTIONS MADE TO STA. POSITION ONLY"
GO TO CREKT
GO TO CUMBAC
734
J = LSTCNT ** 2
738
SQMA ARRAY → SQMC
739
SQBA ARRAY → SQBC
LSTCNT -6
≤0
>0
GO TO 12134
12135
M = 7
N = LSTCNT
K = 6 * LSTCNT
740
741
CLEAR 1ST 6 ROWS OF MATRIX EXCEPT 1ST 6 ELEM
M = M+LSTCNT
N = N+LSTCNT
N-K
≤0
>0
742
M = K+1
743
CLEAR REMAINING ROWS OF MATRIX
M = K+7
K = LSTCNT+1
C


C
744
745
SET DIAGONAL ELEMENTS AFTER 1ST 6 TO 1.0
CLEAR MATRIX B AFTER 1ST 6 ELEM.
12134
LMΦDE = 3
≠0
MCΦDE (3)
=0
746
747
748
RESTORE MATRIX
SQMC → QLSQA
SQBC → QLSQB
GO TO LSQS
737
WRITE: "CORRECTIONS MADE TO ORB. ELEM ONLY"
GO TO EREXT
750
GO TO CUMBAC

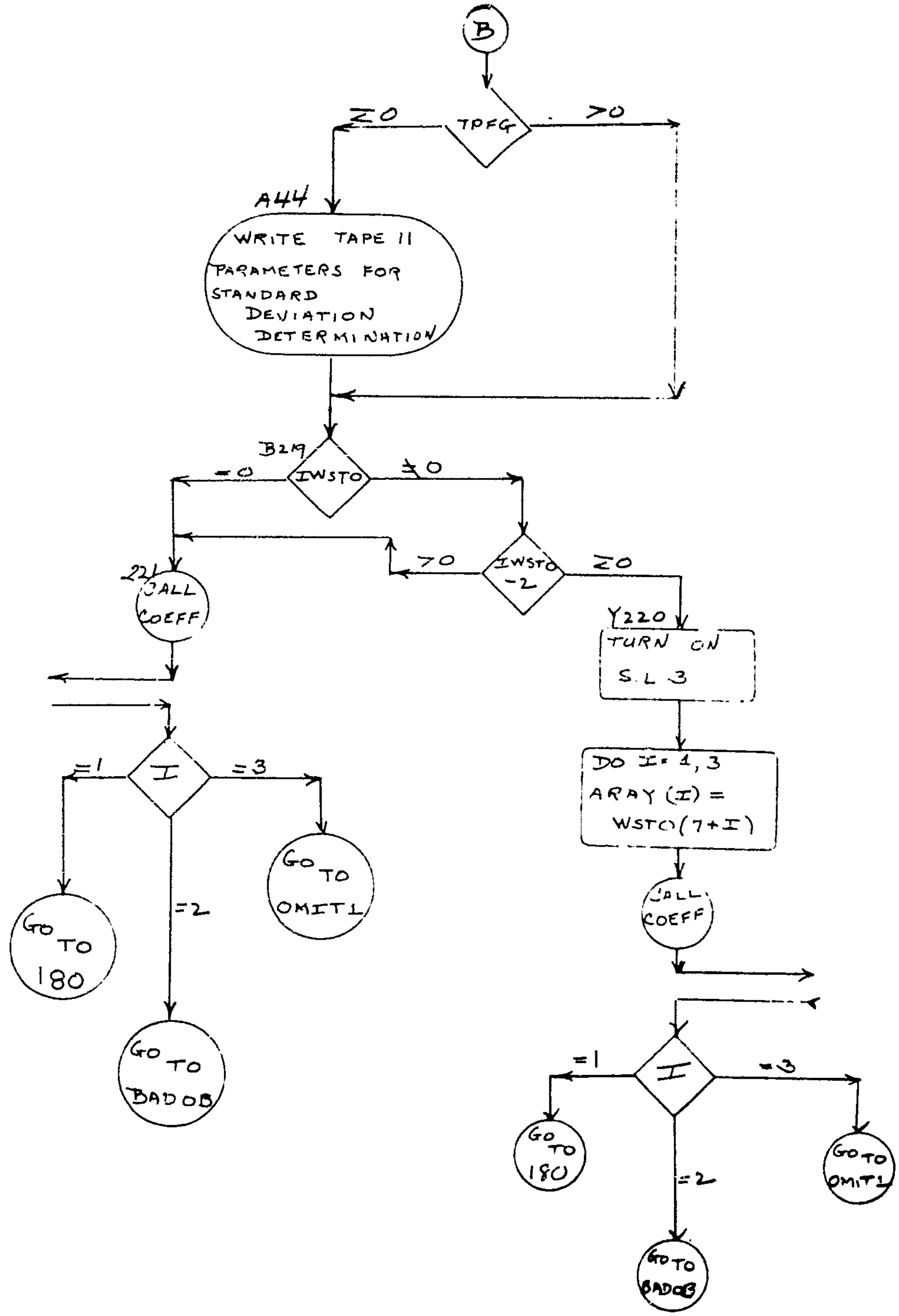

MOVOB
READ TAPE 9 (WSTØ(I), I=1,16), ASIG1, IAXD, SPHA, CPHA, SDEL, CCDEL, COSTH, SINTH, IBMUTH.
I=IXXX
105
NTAPE3
≥0
>0
I=NCNTRL
<0
≥0
PASIEKA
RW1
READ TAPE 3 INTO EPHEM (J)
RW2
WSTO(4) → PASIEKA
LSTAD=J
J=LSTAD,I,8
A
DT
≥0
<0
CTV3
WSTO(4) − EPHEM(J)
CTV2
EPHEM(J) − WSTO(4)
70
GO TO 203
NTAPE3 −1
=0
<0
X201
I=NCNTRL
TSTER
WRITE: ERROR IN TESTING TIME POINTS
EXIT
201
202
DΦ J=1,24
KK=IXXX−24+J
XXXX(J)=EPHEM(KK)
READ TAPE 3 INTO EPHEM
NTAPE3=NTAPE3−1
ITSTAD=ITSTAD+1
LSTAD=1

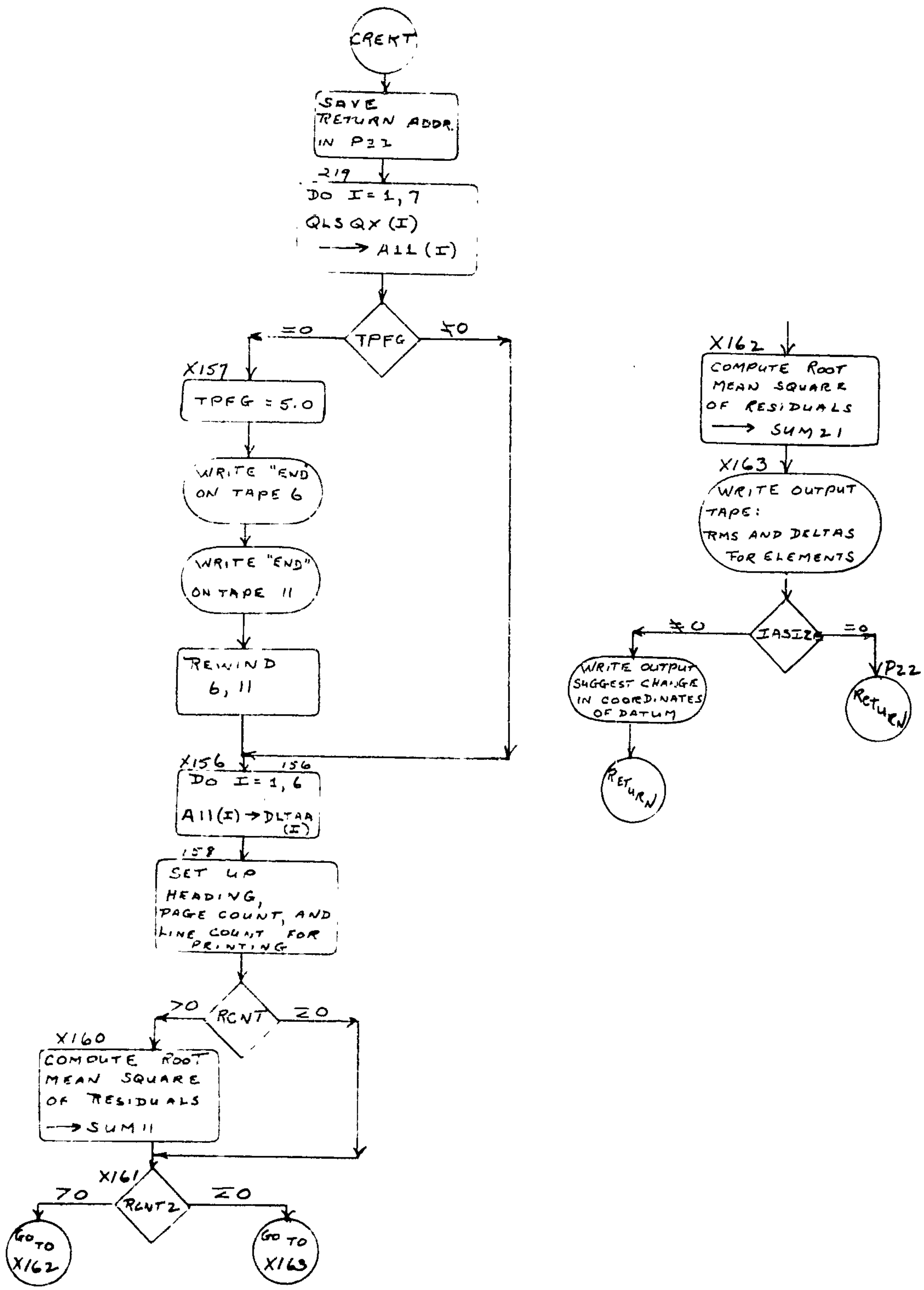
OBSERVATION TIME FOUND

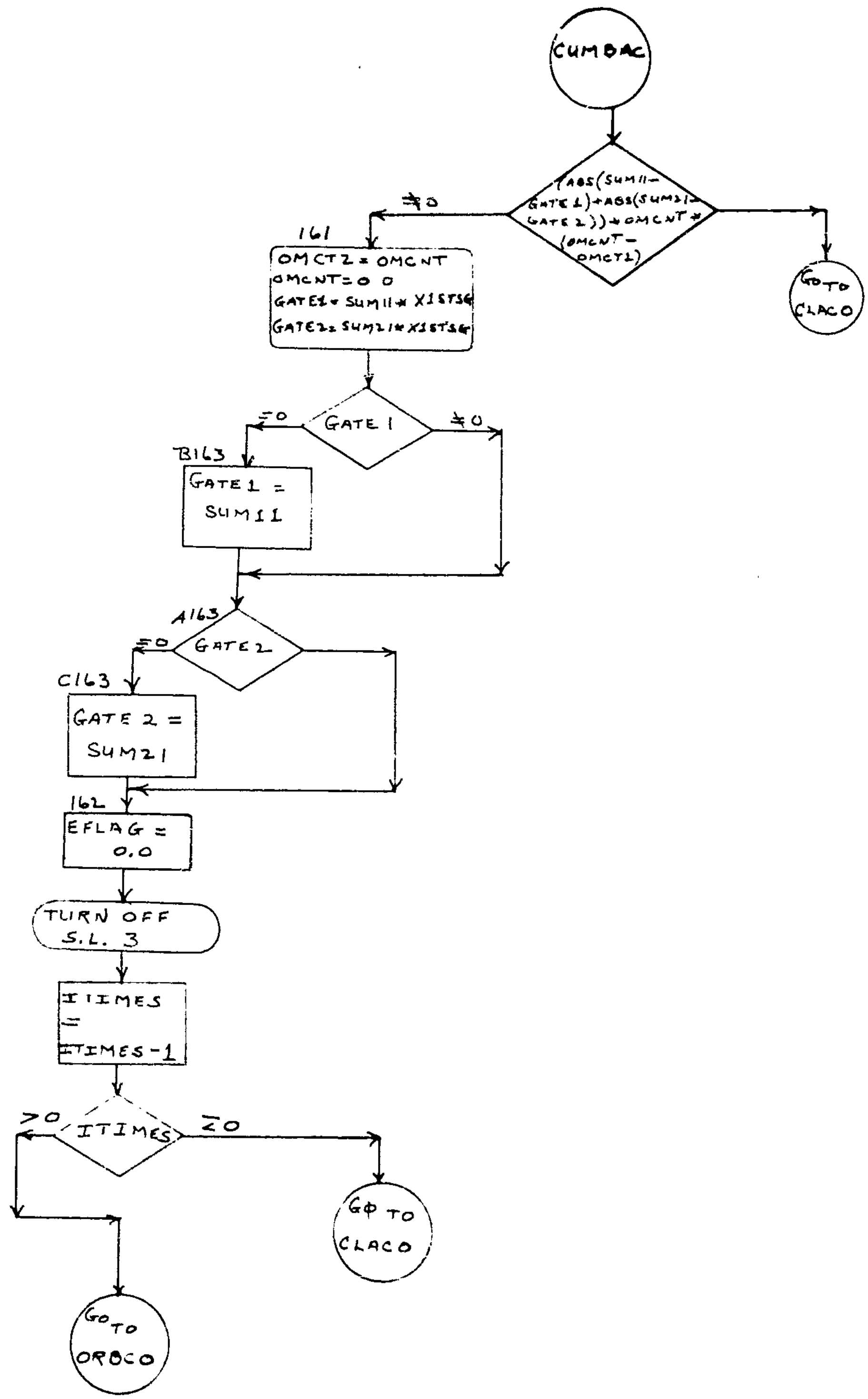




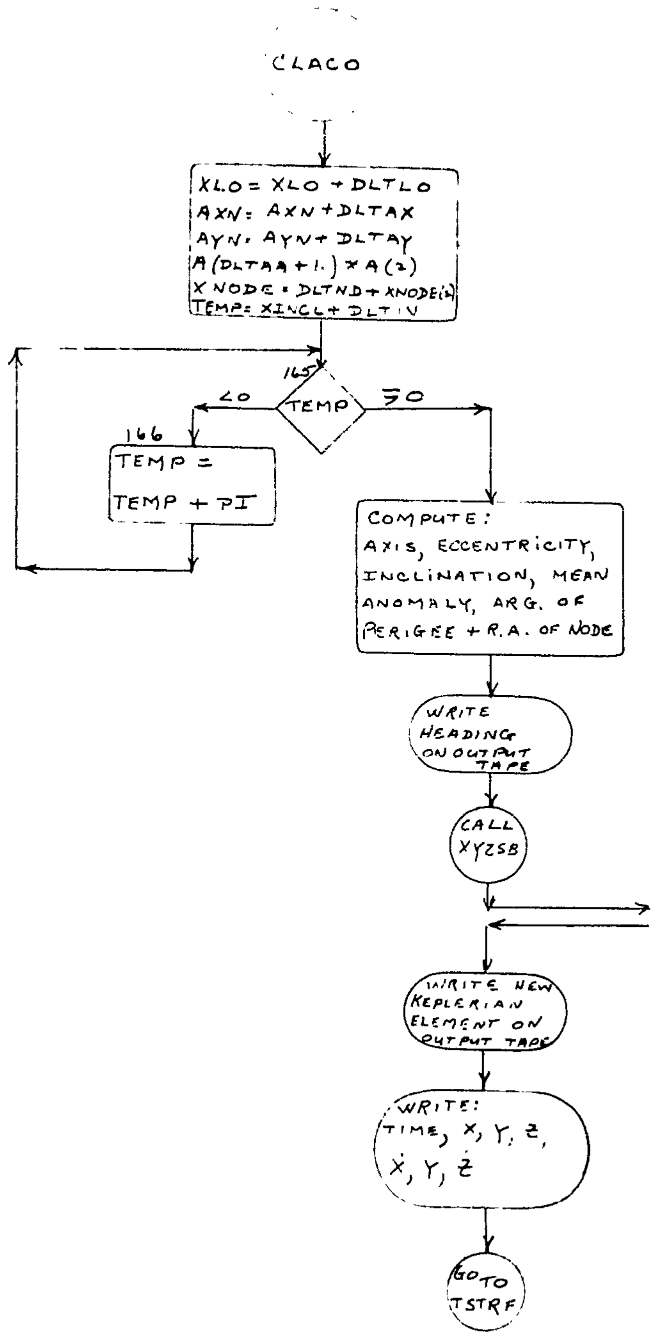
A
TPFG
≤0
>0
B1969
WRITE TAPE 6
TIME, X, Y, Z, Ẋ, Ẏ, Ż
A1970
XNUZ = ZDOT
A2OR = AR * A
RSUBU = A2OR * ESINE
UMUO = (U - UO) * (-3/2)
RSUBA = (UMUO * RSUBU) + R
ROVA = R/A
RXN = (AXNI - COSEO) * A2OR
RYN = (AYNI - SINEO) * A2OR
USUBU = RTESQ * A2OR
USUBA = USUBU * UMUO
DENOM = XLGR^2 * RTESQ
TEMP = (XLGR * ECOSE) * (-1.) - ESQ + (1.)
UYN = (TEMP/DENOM) * ESINE
UXN = UYN * AXNI
ROVA = 1.0 + ROVA
UXN = UXN + ROVA * SINEO
UXN = ((-AYNI/XLGR) + UXN) * A2OR
UYN = UYN * AYNI - ROVA * COSEO
UYN = A2OR * ((AXNI/XLGR) + UYN
RHOC = 0.
DO K = 1,3
RHOX(K) = WSTO(4+K) + X(K)
RHOC = RHOC + RHOX(K)^2
RHOC = √RHOC
XLX(K) = RHOX(K)/RHOC
DELTX(K) = WSTO(K+13) - XLX(K)
B

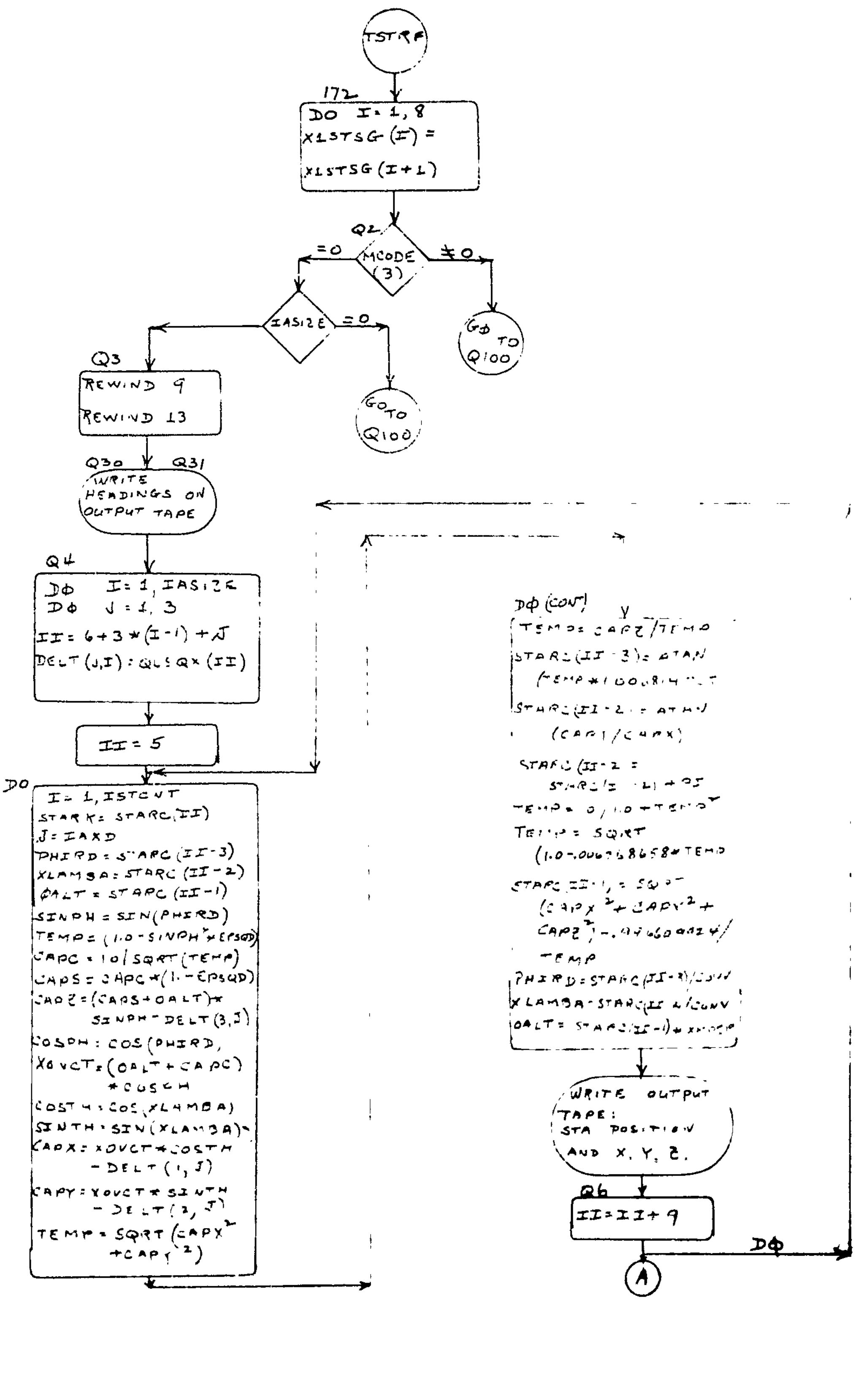

B
TPFG
≤0
>0
A44
WRITE TAPE 11 PARAMETERS FOR STANDARD DEVIATION DETERMINATION
B219
IWSTO
=0
≠0
IWSTO -2
>0
≤0
221
CALL COEFF
Y220
TURN ON S.L 3
DO I = 1, 3 ARAY (I) = WSTO(7+I)
CALL COEFF
I
=1
=2
=3
GO TO 180
GO TO BADOB
GO TO OMIT1
I
=1
=2
=3
GO TO 180
GO TO BADOB
GO TO OMIT1

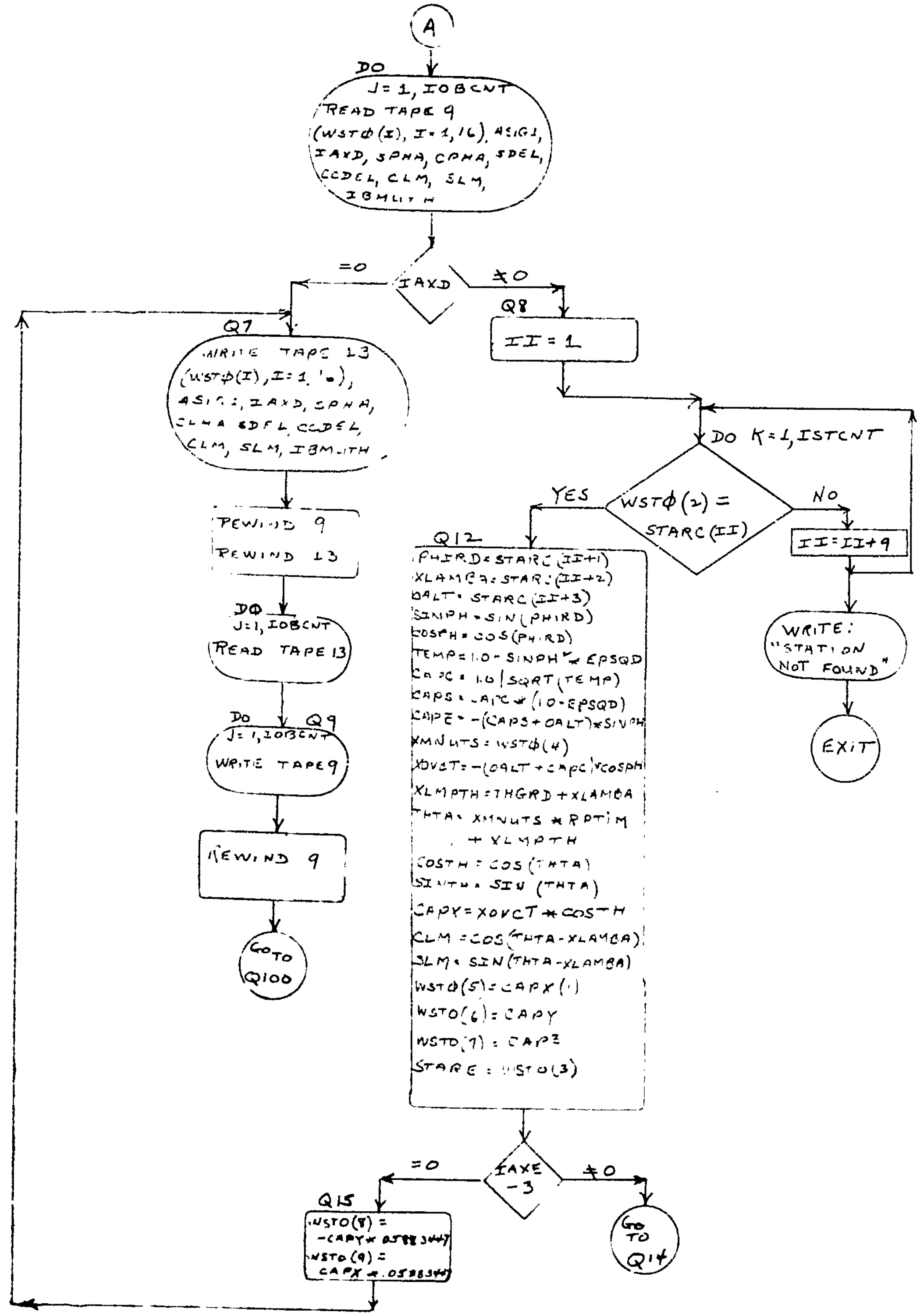

CREKT
SAVE RETURN ADDR. IN P22
219
DO I=1,7 QLSQX(I) ⟶ A11(I)
=0
TPFG
≠0
X157
TPFG = 5.0
WRITE "END" ON TAPE 6
WRITE "END" ON TAPE 11
REWIND 6, 11
X156
156
DO I=1,6 A11(I) → DLTAA(I)
158
SET UP HEADING, PAGE COUNT, AND LINE COUNT FOR PRINTING
>0
RCNT
≥0
X160
COMPUTE ROOT MEAN SQUARE OF RESIDUALS ⟶ SUM11
X161
>0
RCNT2
≥0
GO TO X162
GO TO X163
X162
COMPUTE ROOT MEAN SQUARE OF RESIDUALS ⟶ SUM21
X163
WRITE OUTPUT TAPE: RMS AND DELTAS FOR ELEMENTS
≠0
IASIZE
=0
WRITE OUTPUT SUGGEST CHANGE IN COORDINATES OF DATUM
RETURN
P22
RETURN



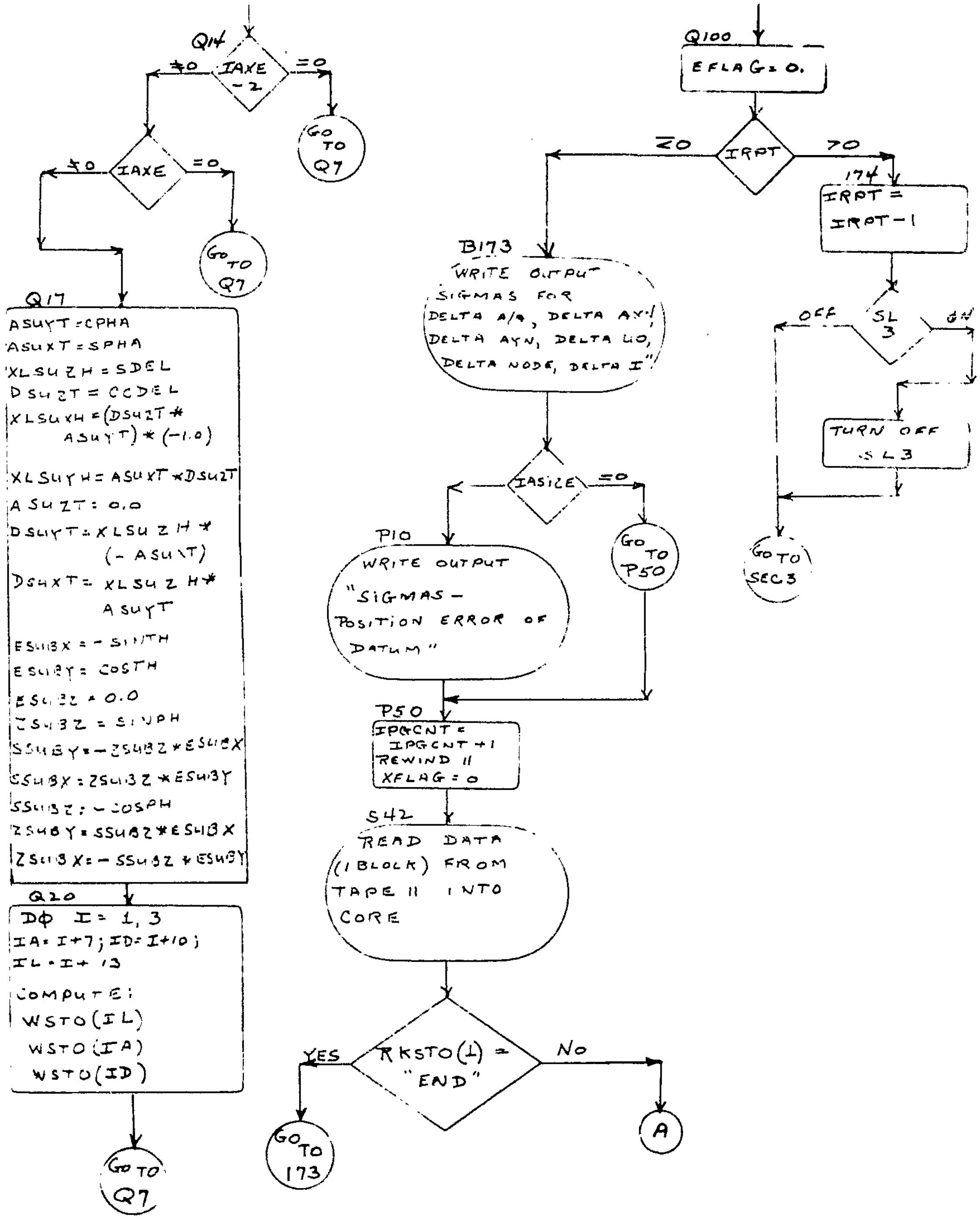
CUMBAC
ABS(SUM11-GATE1)+ABS(SUM21-GATE2))+OMCNT*(OMCNT-OMCT2)
≠0
GO TO CLACO
161
OMCT2 = OMCNT
OMCNT=0.0
GATE1 = SUM11*X1STSG
GATE2 = SUM21*X1STSG
GATE 1
=0
≠0
B163
GATE1 = SUM11
A163
GATE2
=0
C163
GATE 2 = SUM21
162
EFLAG = 0.0
TURN OFF S.L. 3
ITIMES = ITIMES-1
ITIMES
>0
≤0
GO TO CLACO
GO TO ORBCO

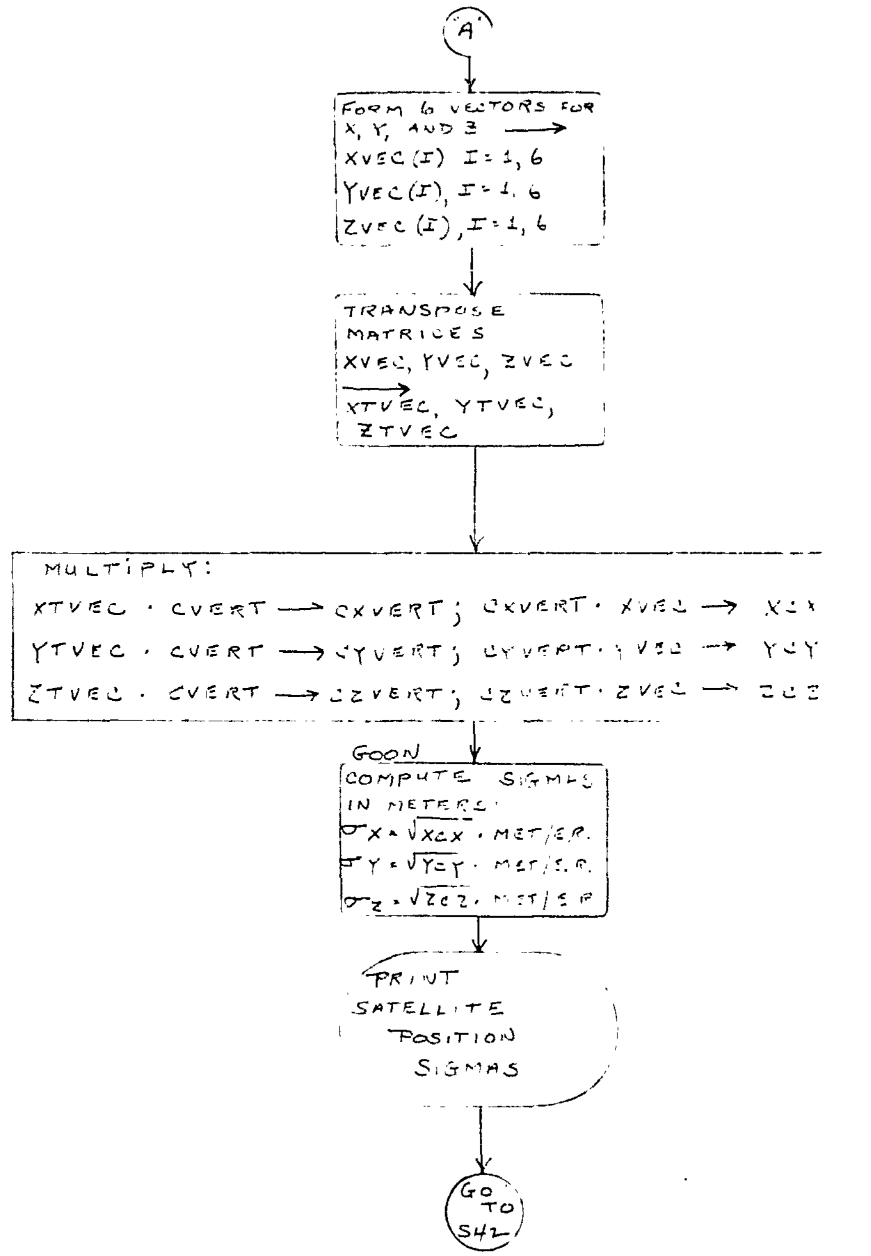

CLACO
XLO = XLO + DLTLO
AXN = AXN + DLTAX
AYN = AYN + DLTAY
A (DLTAA + 1.) * A (2)
XNODE = DLTND + XNODE(2)
TEMP = XINCL + DLTIV
165
TEMP
<0
≥0
166
TEMP = TEMP + PI
COMPUTE:
AXIS, ECCENTRICITY, INCLINATION, MEAN ANOMALY, ARG. OF PERIGEE + R.A. OF NODE
WRITE HEADING ON OUTPUT TAPE
CALL XYZSB
WRITE NEW KEPLERIAN ELEMENT ON OUTPUT TAPE
WRITE: TIME, X, Y, Z, Ẋ, Ẏ, Ż
GO TO TSTRF

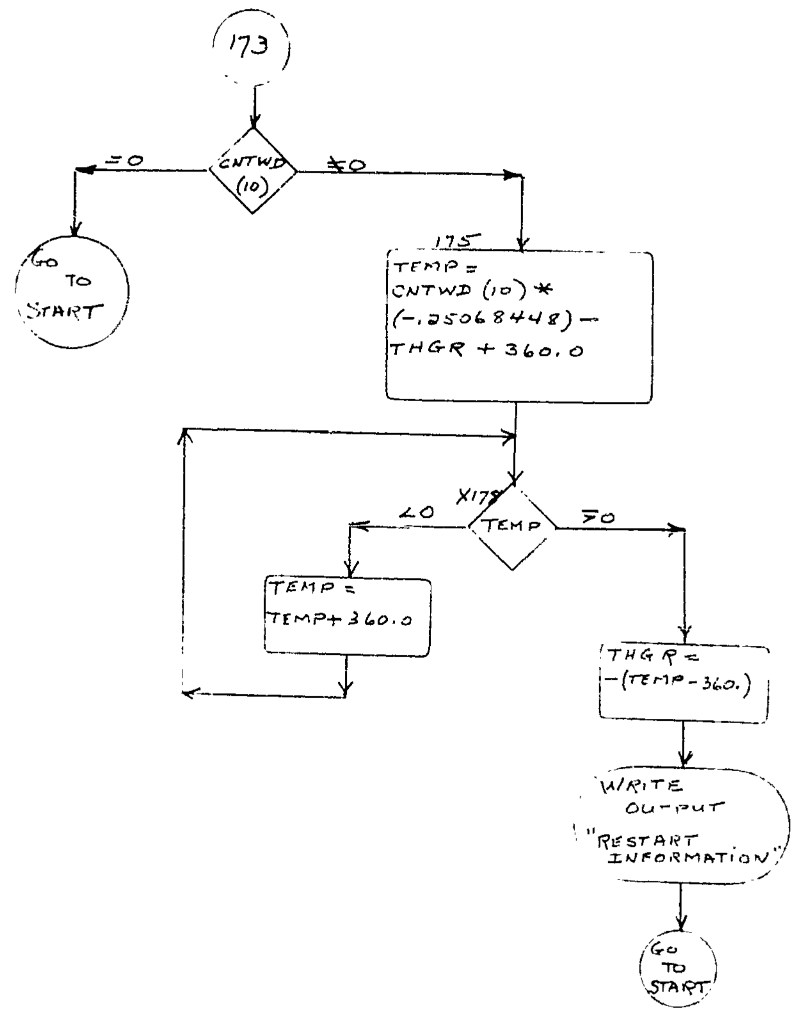

TSTRF
172
DO I = 1, 8
X1STSG(I) = X1STSG(I+1)
Q2
MCODE (3)
= 0
≠ 0
GO TO Q100
IASIZE
= 0
GO TO Q100
Q3
REWIND 9
REWIND 13
Q30
Q31
WRITE HEADINGS ON OUTPUT TAPE
Q4
DΦ I = 1, IASIZE
DΦ J = 1, 3
II = 6+3*(I-1)+J
DELT (J,I) = QLSQX (II)
II = 5
DO
I = 1, ISTCNT
STARK = STARC(II)
J = IAXD
PHIRD = STARC (II-3)
XLAMBA = STARC (II-2)
OALT = STARC (II-1)
SINPH = SIN(PHIRD)
CAPC = 1.0/SQRT(TEMP)
CAPS = CAPC*(1.-EPSQD)
COSPH = COS(PHIRD)
SINTH = SIN(XLAMBA)
DΦ (CONT)
TEMP = CAPZ/TEMP
WRITE OUTPUT TAPE:
STA POSITION AND X, Y, Z.
Q6
II = II + 9
DΦ
A



A
DO
J = 1, IOBCNT
READ TAPE 9
(WSTϕ(I), I = 1, 16), ASIG1,
IAXD, SPHA, CPHA, SDEL,
CCDEL, CLM, SLM,
IBMUTH
=0
IAXD
≠0
Q7
WRITE TAPE 13
(WSTϕ(I), I = 1, 16),
ASIG1, IAXD, SPHA,
CPHA, SDEL, CCDEL,
CLM, SLM, IBMUTH
REWIND 9
REWIND 13
DO
J=1, IOBCNT
READ TAPE 13
DO
Q9
J = 1, IOBCNT
WRITE TAPE 9
REWIND 9
GO TO Q100
Q8
II = 1
DO K = 1, ISTCNT
YES
WSTϕ(2) = STARC(II)
NO
II = II + 9
WRITE: "STATION NOT FOUND"
EXIT
Q12
PHIRD = STARC(II+1)
XLAMBA = STARC(II+2)
OALT = STARC(II+3)
SINPH = SIN(PHIRD)
COSPH = COS(PHIRD)
TEMP = 1.0 - SINPH * EPSQD
CAPC = 1.0 / SQRT(TEMP)
CAPS = CAPC * (1.0 - EPSQD)
CAPZ = -(CAPS + OALT) * SINPH
XMNUTS = WSTϕ(4)
XOVCT = -(OALT + CAPC) * COSPH
XLMPTH = THGRD + XLAMBA
THTA = XMNUTS * RPTIM + XLMPTH
COSTH = COS(THTA)
SINTH = SIN(THTA)
CAPY = XOVCT * COSTH
CLM = COS(THTA - XLAMBA)
SLM = SIN(THTA - XLAMBA)
WSTϕ(5) = CAPX
WSTO(6) = CAPY
WSTO(7) = CAPZ
STARE = WSTO(3)
=0
IAXE - 3
≠0
Q15
WSTO(8) = -CAPY * .0588344
WSTO(9) = CAPX * .0588344
GO TO Q14


Q14
IAXE -2
≠0
=0
GO TO Q7
IAXE
≠0
=0
GO TO Q7
Q17
ASUYT=CPHA
ASUXT=SPHA
XLSUZH=SDEL
DSUZT=CCDEL
XLSUXH=(DSUZT* ASUYT)*(-1.0)
XLSUYH=ASUXT*DSUZT
ASUZT=0.0
DSUYT=XLSUZH*(-ASUXT)
DSUXT=XLSUZH* ASUYT
ESUBX=-SINTH
ESUBY=COSTH
ESUBZ=0.0
ZSUBZ=SINPH
SSUBY=-ZSUBZ*ESUBX
SSUBX=ZSUBZ*ESUBY
SSUBZ=-COSPH
ZSUBY=SSUBZ*ESUBX
ZSUBX=-SSUBZ*ESUBY
Q20
DΦ I=1,3
IA=I+7; ID=I+10;
IL=I+13
COMPUTE:
WSTO(IL)
WSTO(IA)
WSTO(ID)
GO TO Q7
Q100
EFLAG=0.
IRPT
≤0
>0
174
IRPT = IRPT-1
SL 3
OFF
ON
TURN OFF SL3
GO TO SEC3
B173
WRITE OUTPUT SIGMAS FOR DELTA A/a, DELTA AXN, DELTA AYN, DELTA LO, DELTA NODE, DELTA I
IASIZE
=0
GO TO P50
P10
WRITE OUTPUT "SIGMAS - POSITION ERROR OF DATUM"
P50
IPGCNT = IPGCNT+1
REWIND 11
XFLAG=0
S42
READ DATA (1 BLOCK) FROM TAPE 11 INTO CORE
RKSTO(1) = "END"
YES
NO
GO TO 173
A


A
FORM 6 VECTORS FOR
X, Y, AND Z
XVEC(I) I=1, 6
YVEC(I), I=1, 6
ZVEC (I), I=1, 6
TRANSPOSE
MATRICES
XVEC, YVEC, ZVEC
XTVEC, YTVEC,
ZTVEC
MULTIPLY:
XTVEC · CVERT → CXVERT; CXVERT · XVEC → XCX
YTVEC · CVERT → CYVERT; CYVERT · YVEC → YCY
ZTVEC · CVERT → CZVERT; CZVERT · ZVEC → ZCZ
GOON
COMPUTE SIGMAS
IN METERS:
σX = √XCX · MET/E.R.
σY = √YCY · MET/E.R.
σZ = √ZCZ · MET/E.R.
PRINT
SATELLITE
POSITION
SIGMAS
GO
TO
S42


173
CNTWD (10)
=0
≠0
GO TO START
175
TEMP = CNTWD (10) * (-.25068448) - THGR + 360.0
X178
TEMP
<0
≥0
TEMP = TEMP+360.0
THGR = -(TEMP-360.)
WRITE OUTPUT "RESTART INFORMATION"
GO TO START

X
GLOSSARY

A. Subroutines Used in Computation

B. Important Mneumonic Codes

C. Table of Program Constants

A. Subroutines Used In Computations

ALTAC Language

AQDRAG	Computes drag term
CALH	Computes height in meters above oblate earth
COEF	Computes the coefficients of the current correction equation and stores them in an array called Terms
GTOBS	Processes an observation card
RCTFY	Applies corrections for atmospheric drag during the intergration of the ephemeris
SETI	Used during COEF routine to store coefficients in proper locations in array
TBINT	Interpolates in a table
XYZSB	Computes intermediate orbit parameters and satellite positions and velocity vectors
BESSEL	Computes Bessel functions used in drag computation

TAC Language

CHGNXN	Establishes size of matrix as NXN. N can be any size from 6 x 6, 9x9, 12x12,....., 24x24
LSQ	Initialices matrix area to zero
LSQR, LSQS1	Builds and fills in Matrix A and Matrix B
LSQS	Inverts matrix, saves diagonals, relocates 6x6 matrix, and then multiplies the inverted Matrix A by Matrix B

ALTAC Input-Output Subroutines

HEAD	Output routine to print headings on output pages
PRINT	Output routine to print ephemeris
PRINTO	Output routine to print x, y, z, SDOT, YDOT, ZDOT
RCARD	Input-output routine to read in observation cards from Tape 0, and print out this input

READER	Input-output routine to read in station cards from Tape 10 and print out this input
STARTER	Input-output routine to read in 5 specification cards from Tape 0, compute LSUBO, AXN, AYN, HX, HY, HZ, and print out the input and computed elements

ALTAC FUNCTIONS

PROPR(X)	Properizes an angle given in radians
APROPR(X)	Properizes an angle given in degrees

B. Glossary of Important Symbols

A	Semi-major axis
A+1	Semi-major axis
ABSMX	Absolute maximum for angle and range residuals
ABMX2	Absolute maximum for range rate residual reject criteria
ALPHA	Minus RA or azimuth in degrees
ASIG1	Std deviation of observed range
ASIG2	Std deviation of observed range rate
ASIG3	Std deviation of observed right ascension or azimuth
ASIG4	Std deviation of observed declination or elevation
AX	X component of a bar
AY	Y component of a bar
AZ	Z component of a bar
AXNI	X comp in orbit plane (on N bar) of a bar
AYNI	Y comp in orbit plane of a bar
AXI	Same as RKSTO+5
AZI	Same as RKSTO+7
CASE	Case number in BCD, T35
CASE2	Case number in BCD, T23
CD	Initial approximation to the drag coefficient
CDO	Empirically-determined constant to evaluate drag coefficient
CHED1	Constant heading array; Time lat; long; altitude
CNTWD	Col 19 of control card 5 in BCD, T47
CNTWD+1	Epoch, day of year, in BCD W 4 trailing blanks
CNTWD	XL, at new epoch (or restart time)
CNTWD+2	Epoch time of day in minutes --- BCD, T47
CNTWD+3	Number of times to repeat correction - T15 - Same as IRPT
CNTWD+4	REFDA in BCD; 4 trailing blanks
CNTWD+5	REFTM in BCD, T47
CNTWD+9	Minutes difference new epoch minus old epoch (0 if no new ep
CNTWD+10	1st character of 2nd card, T15
CNTWD+14	AXN at new epoch (or restart time)
CNTWD+15	AYN at new epoch (or restart time)

CNTWD+16	HX, at new epoch (or restart time)
CNTWD+21-26	Position + vel vectors for sat. for transfer to tape 6
COND	Intermediate drag term; constant for object
CONTS	Intermediate drag term, Constant = (VCO)**3/(4*EPS*SI
COSI	Cosine (Inclination)
COSO	Cosine of right ascension of node
COUNT	Number of iteration steps in ephemeris
D	Reference diameter of vehicle in meters
DELTA	Elevation or declination of observation in degrees
DD	Day of observation --- Floating point
DT	Time interval used in Runge Kutta integration
ECOSE	ECC* Cos (ECC ANOM)
EFLAG	Column 21 of 5th control card (flo pt integer)
EMIS	Emissivity of the satellite
EO1	Eccentric anomaly + argument of perigee
EPHEM	Core buffer for ephemeris
EPSQD	Square of eccentricity of the terrestrial ellipsoid
ESINE	ECC * SIN (ECC ANOM)
ESQ	1-Eccentricity **2
F	Flattening of the earth
FLP25	Neg of rot. rate of earth in degrees per minute
HDNG	Array contains output page heading for entire run
HH	Hour of observation (Floating point)
HMPRM	3rd harmonic of the earths gravitational potential
HXI	Same as RKSTO+8
HYI	Same as RKSTO+9
HZI	Same as RKSTO+10
IBMUTH	Obs code/ = 0 if RA-DEC / = 1 if AZ-ALT
IPGCNT	Current page number
IPNTFL	Print option as ALTAC integer
IRPT	Same as CNTWD+3 (Number of times to repeat correctior
ISTCNT	Number of stations in starc buffer --- ALTAC integer
IXXX	Size of ephemeris buffer
IY1	Year of observation since 1960, T15
LINE	Line count on page
LO	Mean longitude at epoch

MCODE	Col 38 of CN CRD 5 - If NE 0, correct elems + sta position
M1CODE	Col 39 of CN CRD 5 - If NE 0, corr. sta position only
M2CODE	Col 40 of CN CRD 5 - If NE 0, corr. orbital elements only
MONTH	Month of observation ----- ALTAC integer
NCARD	Switch set by RCARD = 2 if no more obs; 1 if obs ready
NCNTRL	Index for next storage of ephemeris data into ephem buffer
NTAPE3	Number of records on tape 3
NTAPE6	Number of records on tape 6
OALT	Station height in earth radii
ONEPI	Empirically determined constant to evaluate drag coefficient
ORGDA	Epoch. Day of year
ORGTM	Epoch time of day in minutes
P	Semi-latus rectum
PHIRD	Station latitude in radians
QTR	Constant = . 25
R	Magnitude of radius vector to satellite
R5	R**5; Fifth power of radius vector of satellite
RADYN	Degrees per radian
RDRDG	Dot product of RDOT and perturbations vector
RECTI	Switch = 1 if recification for drag has occured; = 0 if not
REFDA	Restart, day of year (new epoch for new elements)
REFTM	Restart time of day
RHOO	Atmos. density at surface of the earth in grams per cubic cm.
RKCNT	Control array for Runge-Kutta integration
RKSTO	Current time (since epoch) of this integration step
RKSTO+3	Time interval for Runge-Kutta integration
RKSTO+4	L, Mean longitude at this integration step
RKSTO+5	X comp. of A bar at this integration step
RKSTO+6	Y comp. of A bar at this integration step
RKSTO+7	Z comp. of A bar at this integration step
RKSTO+8	X comp. of H bar, at this integration step
RKSTO+9	Y comp. of H bar, at this integration step
RKSTO+10	Z comp. of H bar, at this integration step

RRDGR	Dot product of R bar and total perturbation vector
RRDOT	Dot product of R bar and R bar dot
RSQR	R**2 Square of radius vector of satellite
RTA	Square root of semi-major axis
RTESQ	SQRT (1-Eccentricity **2)
RTP	Square root of semi-latus rectum; magnitude of H bar
SATEL	Satellite ID on obs card ---- BCD, trailing blanks
SENSE LIGHT 1	Used by XYZSB to indicate when 1st time thru
SENSE LIGHT 2	On if obs on tape 9 in binary format; Off if not
SENSE LIGHT 3	On, we are processing RA-AZ portion of angle obs.
SENSE LIGHT 4	On if ephemeris is on tape 3; Off if not
SENSE LIGHT 5	Used by GTOBS to signal when obs card are all on tape 9
SENSE LIGHT 10	Used internally in CNTRL
SI	Second of observation (Floating point)
SIDRT	(Rot. rate of earth in deg/mean solar day) minus 360 degrees
SIGS	The Stefan-Boltzmann constant
SINI	Sine of inclination
SINO	Sine of right ascension of node
SIXP9	Empirically determined constant to evaluate drag coefficient
STAID	Station ID from obs card ---- BCD, leading zeroes
STARC	Start of station info buffer - Station ID, BCD, T47
STARC+1	Station latitude in radians
STARC+2	Station east longitude in radians
STARC+3	Station height in earth radii
STARC+4	Station datum number
STARC+5-7	Std. dev. of range, R-Rate, RA-AZ, Decl-Elev
SUM	Sum of squares of residuals of range or angle observations
SUM2	Sum of squares of residuals of range-rate observations
TF	Time after epoch to stop ephemeris calculation
THDOT	Rotation of earth in radians per kemin
TO	Time after epoch to begin ephemeris calculation
U	Mean argument of latitude at time, t.
UO	Mean argument of latitude at epoch
UO+1	U, mod 2π
UTIME	Time of observation in minutes
UZSQR	UZ**2; the square of Z component of U bar

VCO3	Cube of speed of satellite in circular orbit with r = 1 earth radii in cent. cubed per sec cubed
WX	X component of W bar
WZ	Z component of W bar --- WX+2
X1STSG	1st reject factor for residuals
X2NDSG	2nd reject factor for residuals
X3RDSG	3rd reject factor for residuals
X4THSG	4th reject factor for residuals
X5THSG	5th reject factor for residuals
X6THSG	6th reject factor for residuals
X7THSG	7th reject factor for residuals
X8THSG	8th reject factor for residuals
X9THSG	Max no. of times to pass thru obs. before correcting
XBDGR	Partial derivative with respect to bulge of XDOT
XDGR	X component of total perturbations vector (excluding drag)
XINCL	Inclination
XJAY5	5th harmonic of the earth's gravitational potential
XJMPRM	2nd harmonic of the earth's gravitational potential
XK	Drag conversion constant
XK2ER	Meters per earth radii
XKMPRM	4th harmonic of the earth's gravitational potential
XLAMBA	Station east longitude
XLAT	Latitude of sub-satellite point
XLGR	1+ SQRT (1-eccentricity**2)
XLO	Same as LO; mean longitude at epoch
XLONG	East longitude of sub-satellite point
XM	Mass of vehicle in kilograms
XMM	Minute of observation (floating point)
XMNPDA	Constant (minutes per day)
XMPER	Meters per earth radii
XMX	X component of M bar
XMY	Y component of M bar
XMZ	Z component of M bar
XN	Mean motion
XNODE+1	Right ascension of node

XNX	X component of N bar
XNY	Y component of N bar
XOVR5	$(X/R)^5$
YBDGR	Partial derivative of Y dot with respect to bulge; same as XBDGR+1
YDGR	Y component of total perturbations vector; same as XDGR+1
ZBDGR	Partial derivative of Z dot with respect to bulge; same as XBDGR+2
ZDGR	Z component of total perturbations vector; same as XDGR+2
ZOVR5	$(Z/R)^5$

C. Program Constants

2.0	Initial approx to drag coefficient
.2504742E10	Constant used in drag calculation
.92	Emperical constant used in drag calculation
6.972E9	Emperical constnat used in drag calculation
1.173913	Emperical constant used in drag calculation
.0033523299	Flattening
.1685717364E-4	Flattening constant $(3/2F)^2$
.33691870736E-2	Flattening constant $(F+ (3/2F)^2)$
1.62341E-3	2nd harmonic of earth's gravitational potential
-6.E-6	3rd harmonic of earth's gravitational potential
9.09E-6	4th harmonic of earth's gravitational potential
-0.2E-6	5th harmonic of earth's gravitational potential
.001225	Atmospheric density at surface of earth in gms/cm^3
3.141592653	PI
6.283185306	2 PI
57.2957795	Degrees per radian
.01745329251	Radians per degree
1.570796326	π
4.712388978	$3/2\ \pi$
6378273.	Meters per earth radius
3441.605067	Nautical miles per earth radius
.15678527E-6	Earth radii per meter
.15678527	Earth radii per megameter
7.9048	km/sec per earth radius/kemin
.05883447	Rotation rate of earth in radians/kemin
.9856472	Rotation rate of earth in degrees per day minus 360
-.25068448	Negative rotation rate
.9	Emissivity of satellite
.49397574E18	Speed (cubed) of circ satellite at 1 E. R. in meters/second
.5672E-4	Stefan-Boltzmann constant
.07436574	kemins per minute
6.7686580E-3	Square of eccentricity of terrestrial ellipsoid

98.67401	Theta Greenwich for 1960
99.420937	Theta Greenwich for 1961
99.1822167	Theta Greenwich for 1962
98.943500314	Theta Greenwich for 1963
1×10^{-6} E.R.	Sigma for range observations
10.0 cps	Sigma for doppler observations
.00001 radians	Sigma for optical observations
-37922.43	Speed of light

XI

APPENDICES

APPENDIX A

EPHEMERIS COMPUTATION

and

DIFFERENTIAL CORRECTION THEORY*

*From AFCRL 62-892

THEORY

The Orbit Correction Program employs two computational techniques which speed the computation without loss of accuracy:

(1) The satellite motion is numerically integrated by the variation of parameters formulation, thereby eliminating from the integrands the large central acceleration term.
(2) The differential correction uses analytical differential expressions, thereby eliminating the need for more than one numerical integration over the observation period.

By developing the differential equations of motion in terms of parameters which remain invariant in the absence of perturbations to the two-body motion, the dominant two-body term is supressed. The parameters employed in this formulation are:

$\underline{a} = e\underline{P}$	a vector defining eccentricity and perigee location
$\underline{h} = \sqrt{p}\,\underline{W}$	the orbital angular momentum
$L_o = M_o + \omega + \Omega$	the mean longitude of the object at some epoch

These parameters are valid, in the formulation employed here, for all satellite eccentricities and for all inclinations, including zero in either case.

The differential characteristics of a slightly-perturbed satellite orbit are, to a first order, identical to those of the osculating orbit. Thus the cause-and-effect linear relationships needed for differential correction may be developed analytically, rather than by the alternate "variant calculation" procedure where neighboring ephemerides are integrated, in each case with one of the parameters modified by a small amount. The parameters employed in the development of the differential expressions resemble those used in the variation-of-parameters ephemeris program, i.e.:

$\underline{a} = e\underline{P}$	a vector defining eccentricity and perigee location
a	semi-major axis
$U_o = M_o + \omega$	the mean argument of the latitude at some epoch
Ω	nodal longitude
i	inclination

The adoption of Ω and i to describe the orbit plane orientation and of U_o to denote initial position, restricts the development to non-equatorial orbits. In addition, the vector a is described in terms of two components in the orbit plane to avoid redundancy; these components are designated a_{xN} and a_{yN}, with the former in the direction of the ascending node.

Any observation 0_i by an earth-fixed observer may be expressed in terms of these six parameters or elements describing the orbit and the time. First order differential expressions relating observation and parameter follow from the leading term in a Taylor expansion, i.e.,

$$\Delta 0_i = \sum_j \frac{\partial 0_i}{\partial X_j} \Delta X_j \tag{1}$$

where the X_j are the six orbit parameters. Where there are m observations available to define the six parameters, a set of m differential expressions may be written; in matrix form, this set is

$$(\Delta 0_i) = (C_{ij}) (\Delta X_j) \tag{2}$$

where (C_{ij}) is the m x 6 matrix with typical element,

$$C_{ij} = \frac{\partial 0_i}{\partial X_j} \tag{3}$$

The (ΔX_j) is a six component vector, and $(\Delta 0_j)$ is an m component vector.

The elements of the (C_{ij}) matrix are the partial derivatives of the observed quantities with respect to the orbit parameters and may be determined analytically or by variant trajectory calculations, wherein the parameters are varied, one by one, and the resulting changes in the observations are noted.

If there are more observed quantities 0_i than parameters X_j, that is, when $m > 6$, the system is overdetermined and the number of equations may be reduced by the method of least squares. The solution takes the form

$$(\Delta X_j) = \left[(C_{ij})^T (C_{ij}) \right]^{-1} (C_{ij})^T (\Delta 0_i) \tag{4}$$

where the -1 and T superscripts denote inverse matrix and transpose matrix, respectively. The bracketed quantity in (4) is the so-called least square matrix, N:

$$N = (C_{ij})^T \ (C_{ij}) \tag{5}$$

In the interest of achieving efficiency, the C_{ij} are evaluated from analytical expressions, which are detailed in the following section presenting the formulation of the program.

FORMULATION

The Orbit Correction Program can be conveniently divided into three parts, one concerned with the correction process, another concerned with the generation of different types of ephemerides, and, most important, that portion necessary for both, (mainly concerned with the integration of the equations of motion). These parts will be documented below in the inverse order.

VARIATIONS OF THE PARAMETERS

The major portion of the Orbit Correction Program is spent in generating position and velocity of the satellite at successive times by numerically integrating the equations of motion. This integration process is started by generating the initial values of the integrands and other quantities:

Given a_{xN_o}, a_{yN_o}, $\underline{h}_o$, L_o (subscript o indicates epoch), the following procedure is common to both the simulation and differential correction portions of the program:

(a) Compute Greenwich sidereal time at epoch, Θ_{gr}:

$$\Theta_{gr} \text{ (in degrees)} = D\,(\dot{\Theta} - 360^{\circ}) + \dot{\Theta} f + \Theta'_{gr}$$

where D is epoch day number, f is the fraction of a day elapsed from start of epoch day to epoch, Θ'_{gr} is Greenwich sidereal time at the beginning of epoch year, and $\dot{\Theta} = 360\overset{\circ}{.}9856472$, the rotation rate of the earth in degrees per mean solar day.

(b) Compute the semi-latus rectum:

$$p_o = \underline{h}_o \cdot \underline{h}_o$$

(c) Compute the orientation vector, $\underline{W}_o$:

$$\underline{W}_o = \underline{h}_o \Big/ \sqrt{p_o}$$

(d) Compute the orientation angles, i_o, Ω_o, and U_o:

$$\sin i_o = \sqrt{1 - W_{z_o}^{\,2}}$$

$$\cos i_o = W_{z_o}$$

$$i_o = \tan^{-1}\left(\frac{\sin i_o}{\cos i_o}\right) \qquad 0 \leq i_o < \pi$$

$$\sin \Omega_o = W_{x_o} \Big/ \sin i_o$$

$$\cos \Omega_o = -W_{y_o} \Big/ \sin i_o$$

$$\Omega_o = \tan^{-1}\left(\frac{\sin \Omega_o}{\cos \Omega_o}\right) \qquad 0 \leq \Omega_o < 2\pi$$

$$U_o = L_o + \Omega_o \text{ if } W_{z_o} < 0 \quad \text{(retrograde motion)}$$

$$U_o = L_o - \Omega_o \text{ if } W_{z_o} > 0 \quad \text{(direct motion)}$$

(e) Compute the equatorial coordinates of $\underline{a}_o$:

$$\underline{a}_o \begin{cases} a_{x_o} = a_{xN_o} \cos \Omega_o - \cos i_o \, a_{yN_o} \sin \Omega_o \\ a_{y_o} = a_{xN_o} \sin \Omega_o + \cos i_o \, a_{yN_o} \cos \Omega_o \\ a_{z_o} = a_{yN_o} \sin i_o \end{cases}$$

(f) Compute the eccentricity, e_o, and the semi-major axis, a_o:

$$e_o^2 = a_{x_o}^2 + a_{y_o}^2 + a_{z_o}^2$$

$$a_o = p_o / 1 - e_o^2$$

Also common to both simulation and differential correction is the numerical integration. The equations to be integrated are of the form:

$$\frac{dy_i}{dt} = f_i \, (t, y_1, y_2, \ldots, y_6, y_7) \qquad i = 1, 2, \ldots, 7$$

where the y_i equal a_x, a_y, a_z, h_x, h_y, h_z, and L.

The numerical integration scheme used is based on the following fourth order Runge-Kutta method:

$$y_i^{n+1} = y_i^n + \frac{\Delta t}{6} \, (K_{1_i} + 2K_{2_i} + 2K_{3_i} + K_{4_i})$$

where $K_{1_i} = f_i \,(t^n, y_1^n, \ldots y_7^n)$

$$K_{2_i} = f_i \left(t^n + \frac{\Delta t}{2}, y_1^n + \frac{K_{11}}{2}, \ldots, y_7^n + \frac{K_{17}}{2}\right)$$

$$K_{3_i} = f_i \left(t^n + \frac{\Delta t}{2}, y_1^n + \frac{K_{21}}{2}, \ldots, y_7^n + \frac{K_{27}}{2}\right)$$

$$K_{4_i} = f_i \left(t^n + \Delta t, y_1^n + K_{31}, \ldots, y_7^n + K_{37}\right)$$

As can be seen, it is necessary to compute $\frac{d\underline{a}}{dt}$, $\frac{d\underline{h}}{dt}$, and $\frac{dL}{dt}$ from $\underline{a}$, $\underline{h}$, and L, several times at each integration step. The first step is to compute position and velocity, $\underline{r}$, $\dot{\underline{r}}$ from $\underline{a}$, $\underline{h}$, and L.

Given $\underline{a}$, $\underline{h}$, and L at some time t, the following procedure is used to derive position, $\underline{r}$, and velocity, $\dot{\underline{r}}$. Note that whenever the anomalies v, E, and M are used, they appear either in a sum with ω or in products with the coefficient e. Thus, no indeterminacy exists for zero eccentricity. In its present form, precisely equatorial orbits cannot be integrated, since the ascending node is employed as a reference direction.

(a) Compute p, e, a, n:

$$p = \underline{h} \cdot \underline{h} = h_x^2 + h_y^2 + h_z^2$$

$$e^2 = \underline{a} \cdot \underline{a} = a_x^2 + a_y^2 + a_z^2$$

$$a = p/1-e^2$$

$$n = K_e\sqrt{\mu} \big/ a^{3/2} \quad \text{where } K_e\sqrt{\mu} = .07436574$$

(b) Compute the orientation vectors $\underline{W}$, $\underline{M}$, and $\underline{N}$:

$$\underline{W} = \underline{h}/\sqrt{p} \qquad (\text{note } W_z = \cos i)$$

$$M_z = \sqrt{1 - W_z^2} = \sin i$$

$$N_x = -W_y/M_z = \cos \Omega$$

$$M_y = N_x W_z = \cos \Omega \cos i$$

$$N_y = {W_x}/{M_z} = \sin \Omega$$

$$M_x = - N_y W_z = - \sin \Omega \cos i$$

$$N_z = 0$$

(c) Compute the components of $\underline{a}$ in the orbit plane, a_{xN} and a_{yN}:

$$a_{xN} = \underline{a} \cdot \underline{N}$$

$$a_{yN} = \underline{a} \cdot \underline{M}$$

(d) Compute the orientation angles, Ω and U:

$$\Omega = \tan^{-1}\left({N_y}/{N_x} \right) \quad 0 \leq \Omega < 2\pi$$

$$U = L + \Omega \text{ if } W_z < 0 \qquad \text{(retrograde motion)}$$

$$U = L - \Omega \text{ if } W_z > 0 \qquad \text{(direct motion)}$$

(e) Solve Kepler's equation for $E + \omega$ by iteration, using a first guess of U (mod 2π):

$$E + \omega = U + a_{xN} \sin (E + \omega) - a_{yN} \cos (E + \omega)$$

(f) Compute $\underline{r}$ and $\underline{\dot{r}}$:

$$e \cos E = a_{xN} \cos (E + \omega) + a_{yN} \sin (E + \omega)$$

$$e \sin E = a_{xN} \sin (E + \omega) - a_{yN} \cos (E + \omega)$$

$$r = a (1 - e \cos E)$$

$$\dot{r} = \frac{\sqrt{\mu a}}{r} \; e \sin E$$

$$r\dot{v} = \frac{\sqrt{\mu a}}{r} \sqrt{1 - e^2}$$

$$\cos u = \frac{a}{r}\left[\cos (E+\omega) - a_{xN} + a_{yN}\left(\frac{e \sin E}{1 + \sqrt{1-e^2}}\right)\right]$$

$$\sin u = \frac{a}{r}\left[\sin (E+\omega) - a_{yN} - a_{xN}\left(\frac{e \sin E}{1 + \sqrt{1-e^2}}\right)\right]$$

$$\underline{U} = \cos u \, \underline{N} + \sin u \, \underline{M}$$

$$\underline{V} = - \sin u \, \underline{N} + \cos u \, \underline{M}$$

$$\underline{r} = r\underline{U}$$

$$\dot{\underline{r}} = \dot{r}\underline{U} + r\dot{v}\underline{V}$$

From the position and velocity, it is possible to compute the perturbative accelerations, specifically the bulge perturbation, $\dot{\underline{r}}_B$:

$$\dot{x}_B = \frac{x}{r^5} J'(5U_z^2 - 1) + \frac{xz}{r^7} H'(7U_z^2 - 3)$$

$$+ \frac{x}{6r^7} K'(42U_z^2 - 63U_z^4 - 3) + \frac{21xU_z}{8r^8} J_5 a_e^5 \mu (5 - 30 U_z^2 + 33U_z^4)$$

$$\dot{y}_B' = {}^{y}/_{x} \quad \dot{x}_B'$$

$$\ddot{z}_B = \frac{z}{r^5} J' (5U_z^2 - 3) + \frac{3}{5r^5} H' (\frac{35}{3} U_z^4 - 10U_z^2 + 1)$$

$$+ \frac{z}{6r^7} K' (- 63U_z^4 + 70 U_z^2 - 15) + \frac{3}{8r^7} J_5 a_e^5 \mu (-5 + 105U_z^2$$

$$- 315 U_z^4 + 231 U_z^6)$$

where $U_z = \frac{z}{r}$ and $J' = .001\ 623\ 41$

$$H' = -.000\ 006$$

$$K' = .000\ 009\ 09$$

$$J_5 a_e^5 \mu = -.000\ 000\ 2$$

At each point during the integration at which the derivatives dL/dt, d$\underline{h}$/dt, d$\underline{a}$/dt, are evaluated, the perturbative accelerations due to drag $\ddot{x}_D$, $\ddot{y}_D$, $\ddot{z}_D$, must be evaluated and added to the bulge accelerations, $\ddot{x}_B$, $\ddot{y}_B$, $\ddot{z}_B$, to obtain the total perturbative accelerations, $\ddot{x}$, $\ddot{y}$ and $\ddot{z}$.

Given $\underline{r}$ and $\underline{\dot{r}}$, and tabulated values of the density ratio and atmospheric molecular weight versus altitude:

(1) Compute the relative air speed vector,:

$$\gamma_x = \dot{x} + y\,\dot{\theta}$$

$$\gamma_y = \dot{y} - x\,\dot{\theta}$$

$$\gamma_z = \dot{z}$$

Also compute the magnitude of the relative air speed vector

$$\nu = \sqrt{\nu_x^2 + \nu_y^2 + \nu_z^2}\ .$$

$\dot{\theta}$ in the above equations, is the angular rotation rate of the Earth.

(2) Compute the altitude above the oblate spheroid in Earth radii:

$$H = r - 1 - \frac{3}{2} f^2 \left(\frac{z}{r}\right)^4 + \left(f + \frac{3}{2} f^2\right) \left(\frac{z}{r}\right)^2,$$

where f is the flattening of the Earth = 1/298.3.

(3) Look up the density ratio $\sigma = \sigma(H)$ and the molecular weight $M_e = M_e(H)$ from the tabulated data (see Appendix C for the programmed tables), and calculate the atmospheric density :

$$\sigma = \rho / \rho_o$$

where ρ_o is the sea level value of the atmospheric density = 0.001225 gm/cm^3.

(4) Compute the skin temperature of the vehicle:

$$T_s = \left[\frac{\rho C_D (V_{co})^3 \nu^3}{4 \epsilon \sigma_s} + (300)^4 \right]^{1/4}$$

where

σ_s = Stefan-Boltzmann constant = 5.672×10^{-5},

and

ϵ = emissivity of the satellite = .9.

(5) Compute the drag coefficient, C_D, by first computing the auxiliary quantity

$$C = \frac{6.972 \times 10^9 \nu d}{C_{Do} \sqrt{(M_e T_s)}} ,$$

and then

$$C_D = C_{Do} \left(1 + 1.1739130 \, e^{-C\sigma}\right) ,$$

C_{Do} being a reference value of the drag coefficient = 0.92.

(6) Compute the drag terms:

$$\dot{x}_D^{\backslash} = \nu_x \left[C_D \rho \nu \left(- \frac{K \pi d^2}{8m} \right) \right] \qquad x \longrightarrow y,\ z,$$

where d is the diameter of the satellite, m is its mass, and K is a constant relating the units. The program value of $K = 2.504742 \times 10^9$.

Finally we are ready to calculate the integrands $d\underline{a}/dt$, $d\underline{h}/dt$, and dL/dt:

(a) Determine the total perturbations $\underline{\dot{r}}^{\backslash}$

$$\underline{\dot{r}}^{\backslash} = \underline{\dot{r}}_B^{\backslash} + \underline{\dot{r}}_D^{\backslash}$$

(b) Compute:

$$r\,\dot{r}^{\backslash} = \underline{r} \cdot \underline{\dot{r}}^{\backslash}$$

$$\dot{s}\,\dot{s}^{\backslash} = \underline{\dot{r}} \cdot \underline{\dot{r}}$$

$$r\,\dot{r} = \underline{r} \cdot \underline{\dot{r}}$$

and the auxiliary quantities:

$$D = \frac{\underline{r} \cdot \underline{\dot{r}}}{\sqrt{\mu}}$$

$$D^{\backslash} = \frac{\underline{r} \cdot \underline{\dot{r}}^{\backslash}}{\sqrt{\mu}}$$

$$\dot{D}^{\backslash} = \frac{2\underline{\dot{s}} \cdot \underline{\dot{s}}^{\backslash}}{\sqrt{\mu}}$$

(c) Compute $n^{\backslash}$, $r\dot{b}^{\backslash}$, $\ell^{\backslash}$, $\underline{a}^{\backslash}$, $ev^{\backslash}$, and $L^{\backslash}$:

$$n^{\backslash} = -\frac{3}{2}\, na \frac{\dot{D}^{\backslash}}{\sqrt{\mu}}$$

$$r\dot{b}^{\backslash} = \underline{W} \cdot \underline{\dot{r}}^{\backslash}$$

$$\ell^{\backslash} = \frac{z\,(r\dot{b}^{\backslash})}{(1 + W_z)\sqrt{\mu\, p}}$$

$$\underline{a}^{\backslash} = \dot{D}^{\backslash}\underline{r} - D\underline{\dot{r}}^{\backslash} - D^{\backslash}\underline{\dot{r}}$$

$$e\underline{Q} = \underline{W} \times \underline{a}$$

$$-\, e^2 v^{\backslash} = e\underline{Q} \cdot \underline{a}^{\backslash}$$

$$\underline{L}' = -\frac{2D'}{\sqrt{a}} - \frac{e^2 v'}{1 + \sqrt{1 - e^2}}$$

(d) Compute $\underline{h}'$:

$$\underline{h}' = \frac{\underline{r} \times \dot{\underline{r}}'}{\sqrt{\mu}}$$

(e) Compute the derivatives:

$$\frac{dL}{dt} = k_e L' + n$$

$$\frac{d\underline{a}}{dt} = k_e \underline{a}'$$

$$\frac{d\underline{h}}{dt} = k_e \underline{h}'$$

EPHEMERIDES

The word ephemeris is defined as a table of the positions of celestial bodies at regular intervals of time. As used here in reference to the Earth satellite ephemerides, the intervals of time are integration steps. The ephemerides are of three kinds.

The first type of ephemeris shows geocentric position and velocity vectors. No additional formulation is needed to produce this.

The second type of ephemeris shows the position of the satellite as latitude, longitude, and height above the geoid. This is the subsatellite track.

The third type of ephemeris shows acquisition coordinates for a number of designated sensors. These "look angles" show the coordinates at which the satellite would appear to the station if it had started with the given boundary conditions at the epoch. These acquisition coordinates are produced for every integration step at which the satellite is above the horizon of the designated station.

a. <u>Computation of Sub-Satellite Track:</u>

If the option to compute a sub-satellite track consisting of latitude, ϕ, East longitude, λ_E, and height above the earth, H, is chosen, then these quantities are computed at each point of the ephemeris as follows:

$$\phi = \tan^{-1}\left[\frac{U_z}{(1-f)^2\sqrt{1-U_z^{\,2}}}\right] \qquad -90^\circ \leq \phi \leq 90^\circ$$

where $f = \frac{1}{298.3}$ is the flattening of the Earth

$$\lambda_E \text{ (in degrees)} = \theta - \dot{\theta}\,(t-t_o) - \theta_{gr}, \qquad \lambda_E \geq 0^\circ$$

where $\theta = \tan^{-1} (\frac{y}{x})$ $\quad 0^\circ \leq \theta < 360^\circ$

and $\dot{\theta} = .250\ 684\ 48$ is the rotation rate of the Earth in degrees per solar minute.

$$H \text{ (in Earth radii)} = r - 1 + (\frac{3}{2} f^2 + f) U_z^2 - \frac{3}{2} f^2 U_z^4$$

b. Simulation of Acquisition Coordinates:

This part of the program simulates "observations" of the satellite in the orbit specified by the input parameters. Given ϕ, the latitude of a station in degrees, λ_E, the East longitude of the station in degrees, H, the height of the station above sea level in meters, and θ_{gr}, Greenwich Sidereal Time at the initial time, t_o, in degrees, the following procedure computes α, δ, A, h, ρ, and $\dot{\rho}$ for time t:

(1) Convert ϕ, λ_E, and θ_{gr} to radians and H to Earth radii

(2) Compute:

$$C = (1 - \epsilon^2 \sin^2 \phi)^{-1/2}$$

where $\epsilon^2 = 2f - f^2$, $f = \frac{1}{298.3}$ = the flattening of the Earth.

$$S = C (1 - \epsilon^2)$$

(3) Compute:

$$\theta = .004\ 375\ 269\ 1\ (t - t_o) + \theta_{gr} + \lambda_E$$

(4) Compute the station vector, $\underline{R}$:

$$X = -(C + H) \cos\phi \cos\Theta$$

$$Y = -(C + H) \cos\phi \sin\Theta$$

$$Z = -(S + H) \sin\phi$$

(5) Compute the slant range ρ:

$$\underline{\rho} = \underline{r} + \underline{R}$$

$$\rho = \sqrt{\underline{\rho} \cdot \underline{\rho}} = \sqrt{(x + X)^2 + (y + Y)^2 + (z + Z)^2}$$

(6) Compute azimuth, A, and elevation angle, h, of object as "seen" from the observation station:

$$L_{xh} = \frac{(x+X) \cos\Theta \sin\phi + (y+Y) \sin\Theta \sin\phi - (z+Z) \cos\phi}{\rho}$$

$$L_{yh} = \frac{-(x+X) \sin\Theta + (y+Y) \cos\Theta}{\rho}$$

$$L_{zh} = \frac{(x+X) \cos\Theta \cos\phi + (y+Y) \sin\Theta \cos\phi + (z+Z) \sin\phi}{\rho}$$

$$h = \tan^{-1}\left(\frac{L_{zh}}{\sqrt{1 - L_{zh}^2}}\right), \quad -\pi/2 \leq h \leq \pi/2$$

If $h < 0$ (i.e., the object is below the local horizon) the output for this time is omitted.

$$A = \tan^{-1}\left(\frac{L_{yh}}{-L_{xh}}\right) \quad 0 \leq A < 2\pi$$

The quadrant is determined from an examination of the sign of the numerator and denominator.

(7) Compute topocentric right ascension, α, and declination, δ:

$\underline{L} \quad (L_x, L_y, L_z) = \underline{\rho}/\rho$

$$\delta = \tan^{-1}\left(\frac{L_z}{\sqrt{1-L_z^2}}\right) \qquad -\pi/2 \leq \delta \leq \pi/2$$

$$\alpha = \tan^{-1}\left(\frac{L_y}{L_z}\right) \qquad 0 \leq \alpha < 2\pi$$

The quadrant is determined from sign of numerator and denominator.

(8) Compute the slant range rate, $\dot{\rho}$:

$$\left.\begin{aligned} \dot{X} &= -Y\dot{\theta} \\ \dot{Y} &= X\dot{\theta} \\ \dot{Z} &= 0 \end{aligned}\right\} \dot{\theta} = 0.058\ 834\ 47$$

$$\dot{\underline{\rho}} = \dot{\underline{r}} + \dot{\underline{R}} \quad (\dot{X}, \dot{Y}, \dot{Z})$$

$$\dot{\rho} = \underline{L} \cdot \dot{\underline{\rho}} = L_x \ (\dot{x} + \dot{X}) + L_y \ (\dot{y} + \dot{Y}) + L_z \ (\dot{z} + \dot{Z})$$

DIFFERENTIAL CORRECTION OF ORBITAL ELEMENTS:

This part of the program relates residuals in the observations at time, t, to corrections to be applied to the initial orbital parameters at time t_o.

The procedure calculates the orbital parameters and quantities associated with the station coordinates at the observation time. It combines these quantities to obtain the coefficients of the linear relationships relating residuals to any combination of $\Delta a_o / a_o$, Δa_{xN_o}, Δa_{yN_o}, ΔU_o, $\Delta \Omega_o$, Δi_o and represents the observations to determine the residuals. Finally, the corrections to be applied to the parameters are determined by solving the (usually overdetermined) system of linear correction equations.

a. Forming the Linear Correction Equations

Given:

(1) ϕ, the latitude in degrees

(2) λ_E, the longitude in degrees

(3) H, meters above sea level

of the observing station

(4) θ_{gr_o} Greenwich Sidereal time at epoch in degrees

(5) t, the time of the observation in minutes, and

(6) one or more observed quantities at this time,

the following will compute one line of the above-mentioned system for each observed quantity.

(1) Repeat steps 1 through 4 under simulation to obtain θ and ϕ in radians and the station vector $\underline{R}$.

(2) Compute the coefficients R and U where:

$$R_u = \left(a^2/r\right) e \sin E$$

$$R_a = r - \frac{3}{2} (U - U_o) R_u$$

$$R_{xN} = \left(a^2/r\right) \left[a_{xN} - \cos (E + \omega)\right]$$

$$R_{yN} = \left(a^2/r\right) \left[a_{yN} - \sin (E + \omega)\right]$$

$$U_u = \left(a^2/r\right)\sqrt{1 - e^2}$$

$$U_a = - \frac{3}{2} (U - U_o) U_u$$

$$U_{xN} = \frac{a^2}{r} \left\{ (1 + \frac{r}{a}) \sin (E+\omega) + a_{xN}\, e \sin E \right.$$

$$\left. \left[\frac{e^2 - (1 + \sqrt{1 - e^2}) e \cos E}{\sqrt{1 - e^2}\,(1 + \sqrt{1 - e^2})^2} \right] - \frac{a_{yN}}{1 + \sqrt{1 - e^2}} \right\}$$

$$U_{yN} = \frac{a^2}{r} \left\{ - (1 + \frac{r}{a}) \cos (E +\omega) + a_{yN}\, e \sin E \right.$$

$$\left. \left[\frac{e^2 - (1 + \sqrt{1 - e^2}) e \cos E}{\sqrt{1 - e^2}\,(1 + \sqrt{1 - e^2})^2} \right] + \frac{a_{xN}}{1 + \sqrt{1 - e^2}} \right\}$$

(3) Compute:

$$\underline{\rho}_c = \underline{r} + \underline{R}$$

$$\rho_c = \sqrt{\underline{\rho}_c \cdot \underline{\rho}_c}$$

$$\underline{L}_c = \frac{\underline{\rho}_c}{\rho_c}$$

(4) If ρ, the slant range is observed, compute

$$\Delta\rho = \rho - \rho_c$$

Form the coefficients:

$$C_{\frac{\Delta a}{a}} = (\underline{L}_c \cdot \underline{U})\, R_a + (\underline{L}_c \cdot \underline{V})\, U_a$$

$$C_{\Delta a_{xN}} = (\underline{L}_c \cdot \underline{U})\, R_{xN} + (\underline{L}_c \cdot \underline{V})\, U_{xN}$$

$$C_{\Delta a_{yN}} = (\underline{L}_c \cdot \underline{U})\, R_{yN} + (\underline{L}_c \cdot \underline{V})\, U_{yN}$$

$$C_{\Delta U_o} = (\underline{L}_c \cdot \underline{U})\, R_u + (\underline{L}_c \cdot \underline{V})\, U_u$$

$$C_{\Delta\Omega} = (\underline{L}_c \cdot \underline{V})\, r \cos i - (\underline{L}_c \cdot \underline{W})\, r \sin i \cos u$$

$$C_{\Delta i} = (\underline{L}_c \cdot \underline{W})\, r \sin u$$

Enter the following linear correction equation into the system of such equations:

$$\Delta\rho = C_{\frac{\Delta a}{a}} \frac{\Delta a_o}{a_o} + C_{\Delta a_{xN}} \Delta a_{xN_o} + C_{\Delta a_{yN}} \Delta a_{yN_o}$$

$$+ C_{\Delta U_o} \Delta U_o + C_{\Delta\Omega} \Delta\Omega_o + C_{\Delta i} \Delta i_o$$

(5) If A, azimuth, and h, elevation angle are observed, compute:

$$\left.\begin{aligned} S_x &= \sin\phi \cos\theta \\ S_y &= \sin\phi \sin\theta \\ S_z &= -\cos\phi \end{aligned}\right\} \underline{S}$$

$$\left.\begin{aligned} E_x &= -\sin\theta \\ E_y &= \cos\theta \\ E_z &= 0 \end{aligned}\right\} \underline{E}$$

$$\left.\begin{aligned} Z_x &= \cos\phi \cos\theta \\ Z_y &= \cos\phi \sin\theta \\ Z_z &= \sin\phi \end{aligned}\right\} \underline{Z}$$

$$\left.\begin{aligned} L_{xh} &= -\cos A \cos h \\ L_{yh} &= \sin A \cos h \\ L_{zh} &= \sin h \end{aligned}\right\} \quad \underset{\sim}{L}_h$$

$$\left.\begin{aligned} A_{xh} &= \sin A \\ A_{yh} &= \cos A \\ A_{zh} &= 0 \end{aligned}\right\} \quad \tilde{\underset{\sim}{A}}_h$$

$$\left.\begin{aligned} D_{xh} &= \cos A \sin h \\ D_{yh} &= -\sin A \sin h \\ D_{zh} &= \cos h \end{aligned}\right\} \quad \tilde{\underset{\sim}{D}}_h$$

$$\underset{\sim}{L}_{obs} = L_{xh}\,\underline{S} + L_{yh}\,\underline{E} + L_{zh}\,\underline{Z}$$

$$\tilde{\underset{\sim}{A}}_{obs} = A_{xh}\,\underline{S} + A_{yh}\,\underline{E} + A_{zh}\,\underline{Z}$$

$$\tilde{\underset{\sim}{D}}_{obs} = D_{xh}\,\underline{S} + D_{yh}\,\underline{E} + D_{zh}\,\underline{Z}$$

Compute $\underline{\Delta L} = \underset{\sim}{L}_{obs} - \underset{\sim}{L}_c$

Form the coefficients as in (4) with $\tilde{\underset{\sim}{A}}_{obs}$ replacing $\underline{L}_c$ and enter the following linear correction equation into the system of such equations:

$$\rho_c \tilde{\underline{A}}_{obs} \cdot \underline{\Delta L} = C_{\frac{\Delta a}{a}} \frac{\Delta a_c}{a_c} + C_{\Delta a_{xN}} \Delta a_{xN_o}$$

$$+ C_{\Delta a_{yN}} \Delta a_{yN_o} + C_{\Delta U_o} \Delta U_o + C_{\Delta \Omega} \Delta \Omega_o + C_{\Delta i} \Delta i_o$$

Again form the coefficients as in (4), this time with $\underline{D}_{obs}$ replacing $\underline{L}_c$, and enter the following linear correction equation into the system of such equations:

$$\rho_c \tilde{\underline{D}}_{obs} \cdot \underline{\Delta L} = C_{\frac{\Delta a}{a}} \frac{\Delta a_o}{a_o} + C_{\Delta a_{xN}} \Delta a_{xN_o}$$

$$+ C_{\Delta a_{yN}} \Delta a_{yN_o} + C_{\Delta U_o} \Delta U_o + C_{\Delta \Omega} \Delta \Omega_o + C_{\Delta i} \Delta i_o$$

(6) If α, topocentric right ascension, and δ, topocentric declination are observed, compute:

$$\left.\begin{aligned} L_x &= \cos\delta \cos\alpha \\ L_y &= \cos\delta \sin\alpha \\ L_z &= \sin\delta \end{aligned}\right\} \quad \underline{L}_{obs}$$

$$\left.\begin{aligned} A_x &= -\sin\alpha \\ A_y &= \cos\alpha \\ A_z &= 0 \end{aligned}\right\} \quad \underline{A}_{obs}$$

$$\left.\begin{aligned} D_x &= -\sin\delta\cos\alpha \\ D_y &= -\sin\delta\sin\alpha \\ D_z &= \cos\delta \end{aligned}\right\} \quad \underline{D}_{obs}$$

Compute $\underline{\Delta L} = \underline{L}_{obs} - \underline{L}_c$

Form the coefficients and compute the linear correction equations as in (5), substituting $\underline{A}_{obs}$ for $\widetilde{\underline{A}}_{obs}$ and $\underline{D}_{obs}$ for $\widetilde{\underline{D}}_{obs}$

(7) If $\dot{\rho}$, the slant range rate, is observed, compute:

$$\left.\begin{aligned} \dot{X} &= -Y\dot{\theta} \\ \dot{Y} &= X\dot{\theta} \\ \dot{Z} &= 0 \end{aligned}\right\} \quad \dot{\underline{R}} \text{ where } \dot{\theta} = 0.058\ 834\ 47$$

$$\dot{\underline{\rho}}_c = \dot{\underline{r}} + \dot{\underline{R}}$$

$$\dot{\rho}_c = \underline{L}_c \cdot \dot{\underline{\rho}}_c$$

$$ex_\omega = a\,(e\cos E - e^2)$$

$$ey_\omega = a\sqrt{1 - e^2}\; e\sin E$$

$$\dot{v} = \frac{r\dot{v}}{r}$$

Compute the coefficients $\dot{R}$ and $\dot{U}$ where:

$$\dot{R}_u = \sqrt{\mu}\, a^{3/2}\; ex_\omega / r^3$$

$$\dot{R}_a = -\dot{r}/2 - 3/2\,(U - U_0)\,\dot{R}_u$$

$$\dot{R}_{xN} = (\sqrt{\mu}\, a^{5/2}/r^3)\left\{ \sin (E+\omega) - a_{xN}\, e \sin E - a_{yN} \right\}$$

$$\dot{R}_{yN} = (\sqrt{\mu}\, a^{5/2}/r^3)\left\{ -\cos (E+\omega) - a_{yN}\, e \sin E + a_{xN} \right\}$$

$$\dot{U}_u = -\sqrt{\mu}\, a^{3/2} e y_\omega / r^3$$

$$\dot{U}_a = \frac{-r\dot{v}}{2} - \frac{3}{2}(U - U_o)\, \dot{U}_u$$

$$\dot{U}_{xN} = (\sqrt{\mu}\, a^{5/2}/r^3)\sqrt{1-e^2}\left\{ \cos (E+\omega) - a_{xN}(1 + \frac{r^2}{ap}) \right\}$$

$$\dot{U}_{yN} = (\sqrt{\mu}\, a^{5/2}/r^3)\sqrt{1-e^2}\left\{ \sin (E+\omega) - a_{yN}(1 + \frac{r^2}{ap}) \right\}$$

(8) Form the coefficients:

$$C_{\frac{\Delta a}{a}} = (\underline{L}_c \cdot \underline{U})\left[\rho_c (\dot{R}_a - \dot{v}\, U_a) - \dot{\rho}_c R_a\right] + (\dot{\rho}_c \cdot \underline{U})\, R_a$$

$$+ (\underline{L}_c \cdot \underline{V})\left[\rho_c (\dot{U}_a + \frac{\dot{r}}{r} U_a) - \dot{\rho}_c U_a\right] + (\dot{\rho}_c \cdot \underline{V})\, U_a$$

$$C_{\Delta a_{xN}} = (\underline{L}_c \cdot \underline{U})\left[\rho_c (\dot{R}_{xN} - \dot{v}\, U_{xN}) - \dot{\rho}_c R_{xN}\right] + (\dot{\rho}_c \cdot \underline{U})\, R_{xN}$$

$$+ (\underline{L}_c \cdot \underline{V})\left[\rho_c (\dot{U}_{xN} + \frac{\dot{r}}{r} U_{xN}) - \dot{\rho}_c U_{xN}\right] + (\dot{\rho}_c \cdot \underline{V})\, U_{xN}$$

$$C_{\Delta a_{yN}} = (\underline{L}_c \cdot \underline{U})\left[\rho_c (\dot{R}_{yN} - \dot{v}\, U_{yN}) - \dot{\rho}_c R_{yN}\right] + (\dot{\rho}_c \cdot \underline{U})\, R_{yN}$$

$$+ (\underline{L}_c \cdot \underline{V})\left[\rho_c (\dot{U}_{yN} + \frac{\dot{r}}{r} U_{yN}) - \dot{\rho}_c U_{yN}\right] + (\dot{\rho}_c \cdot \underline{V})\, U_{yN}$$

$$C_{U_o} = (\underline{L}_c \cdot \underline{U})\left[\rho_c (\dot{R}_u - \dot{v}\, U_u) - \dot{\rho}_c R_u\right] + (\dot{\rho}_c \cdot \underline{U})\, R_u$$

$$+ (\underline{L}_c \cdot \underline{V})\left[\rho_c (\dot{U}_u + \frac{\dot{r}}{r} U_u) - \dot{\rho}_c U_u\right] + (\dot{\rho}_c \cdot \underline{V})\, U_u$$

$$C_{\Delta\Omega} = -(\underline{L}_c \cdot \underline{U})\rho_c\, r\dot{v} \cos i + (\underline{L}_c \cdot \underline{V}) \cos i \left[\rho_c \dot{r} - \dot{\rho}_c r\right]$$

$$+ (\dot{\rho}_c \cdot \underline{V}) r \cos i + (\underline{L}_c \cdot \underline{W}) \sin i \Big[\rho_c\, (r\dot{v} \sin u - \dot{v} \cos u)$$

$$+ \dot{\rho}_c\, r \cos u\Big] - (\dot{\rho}_c \cdot \underline{W})\, r \sin i \cos u$$

$$C_{\Delta i} = (\underline{L}_c \cdot \underline{W})\left[\rho_c\, (r\dot{v} \cos u + \dot{r} \sin u) - \dot{\rho}_c\, r \sin u\right]$$

$$+ (\dot{\rho}_c \cdot \underline{W})\, r \sin u$$

Compute $\Delta\dot{\rho} = \dot{\rho}_{obs} - \dot{\rho}_c$

Enter the following linear correction equation into the system of such equations:

$$\rho_c \Delta\dot{\rho} = C_{\frac{\Delta a}{a}} \frac{\Delta a_o}{a_o} + C_{\Delta a_{xN}} \Delta a_{xN_o} + C_{\Delta a_{yN}} \Delta a_{yN_o} + C_{\Delta U_o} \Delta U_o$$

$$+ C_{\Delta\Omega} \Delta\Omega_o + C_{\Delta i} \Delta i$$

b. <u>Computing the Corrected</u> L_c, a_{xN_o}, a_{yN_o}, h_{x_o}, h_{y_o}, h_{z_o}

Let $\sum_{j=1}^{N} C_{ij} \Delta X_j = \Delta 0_i$, $i = 1, 2, 3, \ldots$ represent all of the linear correction equations (i.e , the C_{ij}'s are the coefficients, the ΔX_j's are the corrections to the orbital parameters at time t_o, the $\Delta 0_i$ being corrected). The following matrix equation is solved to give the corrections, in a least squares sense, to the orbital parameters at time t_o.

$$\begin{bmatrix} \sum c_{i1}^2 & \sum c_{i1} c_{i2} & \cdots & \sum c_{i1} c_{iN} \\ \sum c_{i1} c_{i2} & \sum c_{i2}^2 & \cdots & \sum c_{i2} c_{iN} \\ \vdots & \vdots & & \vdots \\ \sum c_{i1} c_{iN} & \sum c_{i2} c_{iN} & & \sum c_{iN}^2 \end{bmatrix} \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \\ \vdots \\ \Delta x_N \end{bmatrix} = \begin{bmatrix} \sum c_{i1} \ \Delta 0_i \\ \sum c_{i2} \ \Delta 0_i \\ \\ \sum c_{iN} \ \Delta 0_i \end{bmatrix}$$

These corrections are applied as follows (a prime means that the element is a corrected element):

$$a'_{xN_o} = a_{xN_o} + \Delta a_{xN_o}$$

$$a'_{yN_o} = a_{yN_o} + \Delta a_{yN_o}$$

$$a'_o = a_o \left(1 + \frac{\Delta a_o}{a_o}\right)$$

$$\Omega'_o = \Omega_o + \Delta \Omega_o$$

$$i'_o = i_o + \Delta i_o \qquad 0 \leq i_o < \pi$$

$$L'_o = L_o + \Delta U_o - \Delta \Omega_o \text{ if } \cos i'_o < 0$$

$$L'_o = L_o + \Delta U_o + \Delta \Omega_o \text{ if } \cos i'_o > 0$$

$$\underline{W}'_o \begin{cases} W'_{x_o} = \sin i'_o \sin \Omega'_o \\ W'_{y_o} = -\sin i'_o \cos \Omega'_o \\ W'_{z_o} = \cos i'_o \end{cases}$$

$$e_o'^2 = a'^2_{xN_o} + a'^2_{yN_o}$$

$$p_o' = a_o \left(1 - e_o'^2\right)$$

$$\underline{h}_o' = \sqrt{p_o'}\; \underline{W}_o'$$

$$\underline{a}_o' \begin{cases} a'_{x_o} = - a'_{yN_o} \cos i_o' \sin \Omega_o' - a'_{xN_o} \cos \Omega_o' \\ a'_{y_o} = a'_{yN_o} \cos i_o' \cos \Omega_o' + a'_{xN_o} \sin \Omega_o' \\ a'_{z_o} = a'_{yN_o} \sin i_o' \end{cases}$$

Changes made to preceeding ephemeris computation (See Appendix B, C, D).

1. Two subroutines, LUPER and MONEF, have been incorporated into the program to account for lunar perturbations.

2. A subroutine, BELGE, has been added to account for the perturbations due to higher order harmonics in the earth's gravitational potential.

3. A subroutine PRESS accounts for perturbations due to radiatiation pressure.

APPENDIX B

ANALYSIS FOR INCLUSION OF HIGHER ORDER GRAVITY HARMONICS

The earth's gravitational potential is

$$\phi = \frac{\mu}{r} \left\{ \sum_{n=0}^{\infty} \sum_{m=0}^{n} \frac{\left(a_n^m \cos m\lambda + b_n^m \sin m\lambda\right)}{r^n} P_n^m (\sin \phi) \right\}$$

Here

$$\begin{aligned}
\phi &= \text{geocentric latitude} \\
\lambda &= \text{east longitude} \\
P_n^m (x) &= (1-x^2)^{m/2} \frac{d^m}{dx^m} P_n (x) \\
P_n (x) &= \text{Legendre polynomial of degree } n;\ P_n(1) = 1 \\
r &= \text{geocentric radius in earth radii}
\end{aligned}$$

Symmetry considerations give

$$a_1^0 = a_1^1 = b_1^0 = b_1^1 = 0$$

$$a_0^0 = 1$$

The potential accounted for by the original program is

$$\frac{\mu}{r} \left\{ \sum_{n=0}^{5} \frac{a_n^0}{r^n} P_n^0 (\sin \phi) \right\}$$

The remaining potential is

$$\frac{\mu}{r} \sum_{n=2}^{5} \sum_{m=1}^{n} \frac{\left(a_n^m \cos m\lambda + b_n^m \sin m\lambda\right)}{r^n} P_n^m (\sin \phi)$$

$$+ \frac{\mu}{r} \sum_{n=6}^{\infty} \sum_{m=0}^{n} \frac{\left(a_n^m \cos m\lambda + b_n^m \sin m\lambda\right)}{r^n} P_n^m (\sin \phi)$$

The first line of this expression was added to the original potential.

APPENDIX C

ANALYSIS FOR INCLUSION OF LUNAR PERTURBATIONS

The perturbing function of the moon is

$$R = \frac{m}{r_{\leftmoon}}\left[1 + \left(\frac{r}{r_{\leftmoon}}\right)^2 \left(\frac{3}{2} S^2 - \frac{1}{2}\right)\right]$$

where

r = distance of satellite from center of earth, in earth radii

$r_{\leftmoon}$ = distance of moon from center of earth in earth radii

= 384,403/6,371

m = ratio of moon mass to earth mass = $7.35/5.97 \times 10^{-2}$

$S = u_1 v_1 + u_2 v_2 + u_3 v_3$

and

$$u_1 = \cos\phi \cos\lambda$$
$$u_2 = \cos\phi \sin\lambda \qquad \text{(Satellite)}$$
$$u_3 = \sin\phi$$

$$v_1 = \cos\phi_{\leftmoon} \cos\lambda_{\leftmoon}$$
$$v_2 = \cos\phi_{\leftmoon} \sin\lambda_{\leftmoon} \qquad \text{(Moon)}$$
$$v_3 = \sin\phi_{\leftmoon}$$

The moon ephemeris is computed as described in (5). These equations have been truncated to save computer time. The effect of this truncation is minor as far as computing the perturbing acceleration. The values from the truncated version and the full — terms vary around 20 seconds of arc.

$$\ddot{x} = \frac{\partial R}{\partial x}$$

$$\ddot{y} = \frac{\partial R}{\partial y}$$

$$\ddot{z} = \frac{\partial R}{\partial z}$$

APPENDIX D

ANALYSIS FOR INCLUSION OF RADIATION PRESSURE

Form $\underline{U}_{\odot}$, a unit vector from satellite to sun.

$$\underline{U}_{\odot} = \cos \alpha_{\odot} \cos \delta_{\odot} \, \underline{i} + \sin \alpha_{\odot} \cos \delta_{\odot} \, \underline{j} + \sin \delta_{\odot} \, \underline{k}$$

where

a) right ascension of sun, $\alpha_{\odot} = \ell_{\odot} - C_{16} \sin 2\ell_{\odot}$

b) declination of sun, $\delta_{\odot} = C_{17} \sin \alpha_{\odot}$

c) $\ell_{\odot} = L_{\odot} + C_2 (t - t_0) + C_{15} \sin [C_2 (t - t_0) - C_{14}]$

d) $L_{\odot}$ = longitude of sun at Jan 0. 0 of year of interest

e) $C_2 = 360.^{\circ}98564724$

f) C_{14} = (longitude of perigee of sun minus longitude of sun at Jan 0. 0 for year of interest)

g) C_{15} = 2(eccentricity of earth's orbit)

h) $C_{16} = \tan^2$ (mean obliq of ecliptic $\div$ 2)

i) C_{17} = tan (mean obliquity of ecliptic)

Form

$$V = - \frac{A}{m} P \gamma \nu$$

where

A = area of satellite

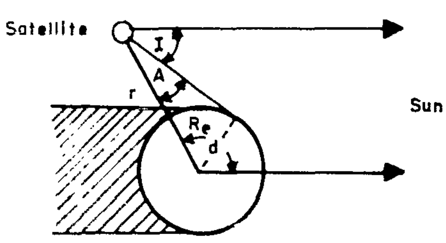
m = mass of satellite

$P = 4.355 \times 10^{-7}$ earth radii/kemin^2

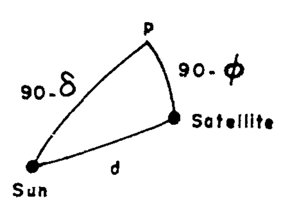
γ = emissivity factor of satellite

ν = eclipse factor

If the satellite is illuminated by the Sun, the angle I must be positive.
ν is 1 if I positive, 0 if I negative.



$I = 180 - d - A$



$$A = \sin^{-1} \left(\frac{Re}{r} \right)$$

$$d = \cos^{-1} (\sin \phi_{\odot} \sin \phi_{\circ} + \cos \phi_{\odot} \cos \phi_{\circ} \cos H)$$

where: $_{\circ}$ denotes satellite

$$H = |RA_{\odot} - RA_{\circ}|$$

Form

$$\grave{\dot{z}} = VU_1$$

$$\grave{\dot{y}} = VU_2$$

$$\grave{\dot{z}} = VU_3$$

and add to the other perturbative accelerations.

APPENDIX E

PROGRAM LISTING

```
         JOB     COMPILE COMPLETED CCC II PROGRAM    29JAN HUSSEY
  RPL    ALTAC2  MAGTAP ,LI
  I      DOC 12 FEB
         IOL IT  T,,T,12,1 ,T PERR,BUFF   TT,10T,12,10,TAPERR,BUF4
                 ETC,5T,,,T PER ,BUF      BT ,3T,,,TAPERR,BUF3
                 HTI ,6T,,,T PERR,BUF     BT , T,,,TAPERR,BUF9
         BT  ,1  ,,,TAPERR,BUF1
         FC ,13 ,12,10, PERR,BUF13
         BT  ,14 ,,,TAPERR,BUF14
         BT  ,15T,,,TAPERR,BUF15$
  T      SAVE    TAPERR,TAPERR,TAPERR $
  T      REFOUT  TAPERR,TAPERR$
         SUBROUTINE  GDRAG(ESQ,E,DRGCO,RATIO,A,Q,BESTO,PERGE,ADRAG,
         GDRAG)
         COMMON  MM  $
         DIMENSION  MMY (4200) $
          DIMENSION Q(2), A(2), BESIO(10)$
         TABLE REF IO(10)      ,IQ(10), ACOFF(9),QCOFF(10)    $
         N=1 M=1         $
          CALL SUMM ( I ,N,M,ES ,ACOEF)$
         M=1  $
         CALL SUMM(I ,N,M,E, COEF)          $
         Q(1)=DRGCO*RATIO *A(1) Q(2)=0.0$
         DO 1I= ,9 $
         TEMP = BESIO(1 -I)* COEF(10-I)+Q(2) $
         IF(I-9) ,3,  $
        2Q(2)=TEMP*E$
        1CONTINUE$
        3ADRAG=(TEMP)*Q(1)* (1) Q(2)=0.0$
         DO  I= ,9 $
         TEMP=BESIO(11- )*QCOEF(11-I)+Q(2)
         IF(I-9) ,6,  $
        5Q(2)=TEMP*E
        4CONTINU $
        6QDRAG=(BESIO(1)*QCOEF+TEMP)*Q(1)*PERGE $
               ETC  $
          START TIC$
  T       0/5T15    $
          0/ T15    $
          0/ 15   $
          0/ 15   $
          0/ 15   $
          0/  15  $
          0/  15  $
          0/ 15   $
          0/ 1T1    $
          0/ 1T1    $
          0/1 T1    $
          0/ 15     $
          0/ 15     $
          0/ 15     $
          0/  15    $
          0/  15    $
          0/ 15     $
          0/  15    
     F    F/ .0$
          F/ .0$
          F/ .0$
          F/ .$
          F/ .0$
          F/ .
```

```
         F/0.0$
         F/.0097656255
         F/.00457763672$
QCOEF    F/0.0$
         F/0.0$
         F/0.0$
         F/0.0$
         F/0.0$
         F/0.0$
         F/0.0$
         F/0.0$
         F/0.0$
         F/-.00108811523$
         PAGE$
         ENDTAC          $
         END$
         SUBROUTINE BELGE(I,TT,W,TH,A,B,C) $
         DIMENSION D(3,3),P(6,6),FR(4),ALFA(6,6),BETA(6,6),T4(3),W(3) $
         DO(ABC) J=1,6  $
         DO(ABC)K=1,6 $
         ALFA(K,J)=0.0 $
         BETA(K,J)=0.0 $
ABC      CONTINUE $
         ALFA(3,3)=1.836E-6$
         ALFA(5,2)=-.6786E-6$
         ALFA(5,3)=.1011E-6$
         ALFA(5,4)=.07425E-6$
         BETA(3,3)=-.9874E-6$
         BETA(5,2)=-.3757E-6$
         BETA(5,3)=.2688E-6$
         BETA(5,4)=-.01569E-6$
   20    CT=COSF(TH)$
         ST=SINF(TH)  $
         X=W(1)*CT+W(2)*ST   $
         Y=-W(1)*ST+W(2)*CT  $
         Z=W(3)    $
         T1=X * X + Y * Y                                        $
         QMU=1.0 $
         RSQ=T1 + Z * Z                                          $
         T2=Z/SQRTF(T1)                                          $
         D(1,2)=-T2* X/RSQ                                       $
         D(2,2)=-T2* Y/RSQ                                       $
         R=SQRTF(RSQ)                                            $
         F=ATANF(T2)                                             $
         C=COSF(F)                                               $
         T=SINF(F)                                               $
         D(3,2)=SQRTF(T1)/RSQ$
         D(1,1)=X/R                                              $
         D(2,1)=Y/R                                              $
         D(3,1)=Z/R                                              $
         D(1,3)=-Y/T1                                            $
         D(2,3)=X/T1                                             $
         D(3,3)=0.                                               $
                                                                 $
                                                                 $
                                                                 $
                                                                 $
                                                                 $
                                                                 $
                                                                 $
                                                                 $
```

```
P(1,1)=1.                                                                    $
P(2,1)=T                                                                     $
P(1,2)= .                                                                    $
P(2,2)= .                                                                    $
DO 1 N=3,6                                                                   $
XN= -1                                                                       $
P(N,N)=(2.* XN-1.)* P( -1,N-1)                                               $
FOR   LEGENDRE POLYNOMIALS                                                   $
P( ,1)=(2.* N-1.)*T*P( -1,1)/XN-(X -1.)*P(N-2,1)/XN                          $
DO 2 M=2,6                                                                   $
XM= -1                                                                       $
L= +1   $
DO 2 N= ,6   $
XN=N-1                                                                       $
P(N,M)=(2.*XN-1.)*T*P(N-1,M)/XN+(2.*XN-1.)*XM*P(N-1,M-1)/XN                  $
IF(N-M-1)3,2,3                                                               $
P(N,M)=P(N,M)-(XN-1.)*P(N-2,M)/XN                                            $
CONTINUE                                                                     $
INSERT THE  COS(PHI) TO THE M TERMS                                          $
Z=1.                                                                         $
DO 4 M=2,6                                                                   $
Z=Z*C                                                                        $
DO 4 N= ,6                                                                   $
P(N,M)=P(N, )*Z                                                              $
FOR  PARTIALS                                                                $
DO 5 N=1,4                                                                   $
FR( )=0.                                                                     $
XL=ATANF(Y/X)                                                                $
IF(XL)6,7,7                                                                  $
XL=XL + 3.141596543                                                          $
XL=PROPR(XL)   $
CL=COSF(XL)                                                                  $
SL=SINF(XL)                                                                  $
U=R                                                                          $
DO 8 N=3,6                                                                   $
XN=N-1                                                                       $
U=U*R                                                                        $
T1=1.                                                                        $
T2=0.                                                                        $
DO 8 M=2,N                                                                   $
T3=T1*CL-T2*SL                                                               $
T2=T1*SL+T2*CL                                                               $
T1=T3                                                                        $
XM= -1 $
FR(4)=FR(4)+(ALFA(N,M)*T1+BETA(N,M)*T2)/(U*R)*P(N,M)$
FR(1)=FR(1)-XN/(U*R)*(ALFA(N,M)*T1+BETA(N,M)*T2)*P(N,M)                      $
FR(3)=FR(3)-(X *ALFA(N,M)*T2-XM*BETA(N,M)*T1)*P(N,M)/U$
FR(2)=FR(2)-(X *P(N,M)*T/C-P(N,M+1))*(ALFA(N,M)*T1+BETA(N,M)*
T2)/U  $
CONTINUE                                                                     $
FR(1)=  U*(FR( )-FR(4))/R$
FR(2)=FR(2)*GM /R $
FR(3)=FR(3)*GM /R $
CALL LUNER(I,TT,F,XL,R,DR,DLA,DLO)                                           $
FR(1)=FR(1)+DR                                                               $
FR(2)=FR(2)+DL                                                               $
FR(3)=FR(3)+DL                                                               $
DO 9 M=1,3                                                                   $
T4( )= .                                                                     $
DO 9 N=1,3                                                                   $
T4( )=  (M)+ ( ,N)*FR( )                                                     $
```

```
        A=T4(1)*CT-T4(2)*ST $
        B=T4(1)*ST+T4(2)*CT $
        C=T4(3)   $
        RETURN
        END (     $
        SUBROUTINE BESSEL (XKDRG,A,E,XKAE,RTKAE,BESIO,BESI1,EMKAE,XKAESQ
            ,   ) $
        EQUIVALENCE (BESI1, BESIO(2) ) $
        DIMENSION Q(2), A(2), BESIO(10), BESI1(1) $
         TABLEDEF XIOCON(12),XI1CON(12),XIOCNS(15),XI1CNS( 15)      $
        XKAE=(A(1)*E)/XKDRG  $
        IF(XKAE-8.0)8,1,1$
       1RTKAE=SQRT (XKAE)$
        TEMP=XIOCON/XKAE$
        DO 3 I=2,12$
        TEMP=TEMP+XIOCON(I)$
        IF(I-12)3,2,2$
       3TEMP=TEMP/XKAE$
       2BESIO(1)=((TEMP)/(RTKAE))*R2PM1 $
        TEMP=XI1CON/XKAE$
        DO 4 I=2,12$
        TEMP=TEMP+XI1CON(I)$
        IF(I-12)5,6,6$
       5TEMP=TEMP/XKAE$
       4CONTINUE$
       6BESI1(1)=(TEMP*R2PM1)/RTKAE$
        GO TO 9
       8EMKAE=EXP (-XKAE)$
        XKAESQ=XKAE**2$
        TEMP=XIOCNS*XKAESQ$
        DO 10I=2,15$
        TEMP=TEMP+XIOCNS(I)$
        IF(I-15)11,12,12$
      11TEMP=TEMP*XKAESQ$
      10CONTINUE$
      12BESIO(1)=TEMP*EMKAE $
*NOTE THAT BESI0=BESI1(-1)$
        TEMP=XI1CNS*XKAESQ$
        DO 13I=2,15$
        TEMP=TEMP+XI1CNS(I)$
        IF(I-15)16,17,17$
      16TEMP=TEMP*XKAESQ$
      13CONTINUE$
      17BESI1(1)=TEMP*EMKAE*XKAE $
       9Q(1)=0.0$
        DO 18I=1,8$
        Q(1)=Q(1)-1.0 $
  18    BESI1(I+1)=BESI1(I)*2.0*Q(1)/XKAE+BESI1(I-1)$
            RETURN$
        START TAC        $
 XIOCON F/551.335894$
        F/110.017145
        F/24.38 5297$
        F/6.07404200$
        F/1.72772755
        F/.57251421$
        F/.22710800 2$
        F/.11215215
        F/.0732421875$
        F/.0703125$
        F/.125$
```

```
       F/1.0$
XI1CON F/-603.844077$
       F/-121.597892$
       F/-27.2488274$
       F/-6.86391429$
       F/-1.99353174$
       F/-6.7659259E-1$
       F/-.277576447$
       F/-.144195557$
       F/-.102539063$
       F/-.1171875$
       F/-.375$
       F/1.0$
XI0CNS F/.490166265E-30$
       F/.384290352E-27$
       F/.259780278E-24$
       F/.14963344E-21$
       F/.0724225848E-18$
       F/.0289690339E-15$
       F/9.38596699E-15$
       F/2.40230755E-12$
       F/.47095028E-9$
       F/.0678168403E-6$
       F/6.78168403E-6$
       F/.434027778E-3$
       F/.015625$
       F/.25$
       F/1.0$
XI1CNS F/.0163388755E-30$
       F/.0137246554E-27$
       F/9.99154915E-27$
       F/6.23472667E-24$
       F/3.29193568E-21$
       F/1.44845170E-18$
       F/.521442611E-15$
       F/.150175472E-12$
       F/.0336393058E-9$
       F/5.65140337E-9$
       F/.678168403E-6$
        F/.0542534723E-3$
       F/2.60416667E-3$
       F/.0625$
       F/.5$
P 82PM1 F/.398942281 $
       END TAC $
       END$
       SUBROUTINE COEFF(ARAY,NTRAP,XMPER)$
*
*     SUBROUTINE COEFF COMPUTES THE COEFFICIENTS OF THE CURRENT
*        CORRECTION EQUATION, AND STORES THEM IN TERMS ARRAY
*
*      ENTER COMMON,DIMENSION, AND EQIV STATEMENTS HERE$
       EQUIVALENCE(RKSTO,RKCNT(2)),(HYI,RKCNT(11)),(HZI,RKCNT(12)),
                  (HXI,RKCNT(10)),(AXI,RKCNT(7)),(AYI,RKCNT(8)),
       (AZI,RKCNT(9)),(AYN,AXN(2)),(HY,HX(2)),(HZ,HX(3)),
       (WY,WX(2)),(WZ,WX(3)),(AY,AX(2)),(AZ,AX(3)),
       (XMY,XMX(2)),(XMZ,XMX(3)),(XNY,XNX(2)),(XNZ,XNX(3)),
       (AYNI,AXNI(2)),(UY,UX(2)),(UZ,UX(3)),(VY,VX(2)),(VZ,VX(3)),
       (Y,X(2)),(Z,X(3)),(YDOT,XDOT(2)),(ZDOT,XDOT(3)),
       (XNUY,XNUX(2)), (XNUZ, XNUX(3)),(EQY  , EQX(2)) ,
       (EQZ, EQX(3)), (COSPH, SINPH(2)), (CAPY, CAPX(2)),
```

```
(CAPZ,CAPX(3)),(RHOY,RHOX(2)),(RHOZ,RHOX(3)),
(X2NDSG,X1STSG(2)),(X3RDSG,X1STSG(3)),(X4THSG,X1STSG(4)),
(X5THSG,X1STSG(5)),(X6THSG,X1STSG(6)),(X7THSG,X1STSG(7)),
(X8THSG,X1STSG(8)),(X9THSG,X1STSG(9)),(ASUBY,ASUBX(2)),
(ASUBZ, SUBX(3)),(DSUBY,DSUBX(2)),(DSUBZ,DSUBX(3)),
(XLSUBY,XLSUBX(2)),(XLSUBZ,XLSUBX(3)),(ESUBY,ESUBX(2)),
(ESUBZ,ESUBX(3)),(SSUBY,SSUBX(2)),(SSUBZ,SSUBX(3)),
(ZSUBY,ZSUBX(2)),(ZSUBZ,ZSUBX(3)),(XLSUYH,XLSUXH(2)),
(XLSUZH,XLSUXH(3)),(ADOTV,ADOTU(2)),(ADOTW,ADOTU(3)),
(ASUYT,ASUXT(2)),(ASUZT,ASUXT(3)),(DSUYT,DSUXT(2)),
(DSUZT,DSUXT(3)),(XLY,XLX(2)),(XLZ,XLX(3)),(DELTY,DELTX(2)),
(DELTZ, DELTX(3)), (IRPT, CNTWD(4)), (IWSTO, WSTO(3)) ,
(DLYV, DLXV(2)), (  DLZV, DLXV(3)) ,
(RXN,RSUBA(2)),(RYN,RSUBA(3)),(RSUBU,RSUBA(4)),(UXN,USUBA(2)),
(UYN,USUBA(3)),(USUBU,USUBA(4)),(RDDTV,RDDTU(2)),(RDDTW,
RDDTU(3)),(RDTXN,RDOTA(2)),(RDTYN,RDOTA(3)),(RDOTV,
RDOTA(4)),(UDTXN,UDOTA(2)),(UDTYN,UDOTA(3)),(UDOTU,UDOTA(4)),
(RDOTU,RDOTA(4)),(IAXD,STARK),
(DGR,CAPD(2)),(DDGR,CAPD(3)) $
EQUIVALENCE(DLTAA(2),DLTAX),(DLTAA(3),DLTAY),(DLTAA(4),DLTLO),
 (DLTAA(5), DLTND), (DLTAA(6), DLTIN) $
COMMON TO, DT, TF, D, XM, THGR, XLO, AXN,
HX,XN,IPGCNT,COUNT,FRAC,P,RTP,WX,SINI,COSI,SINO,
COSO,AX,COND,CONTS,ESQ,E,A,XMX,XNX,AXNI,
XNODE,U,UO,EO1,EO2,SINEO,COSEO,ESINE,ECOSE,R,
SINU,COSU,AR,RTESQ,UX,VX,X,XDOT,RDOT,RTA,RVDOT,
XNUX, XNU , UZFTH, RHO, XNUSQR, TS, C, DC, XOVR5, ZOVR5,
 RSQR, RRDOT, EQX, XLAT, XLONG, RKCNT,
OTPT,CNTWD,STACD,OBSCD,SINPH,CAPD,CAPC,CAPS,
CAPX,OALT,PHIRD,XOVCT,SINTH,COSTH,THTA,TXYZ,
RHOX,RHO1,SMLH,TERMS,WSTO,TIMTB,CSUBI,XLMPTH,XINCL,DLTAA,
RHOC,SUM,ARSDL,XLGR,
RCNT,BSCNT,RESID,EFCNT,LSTAD,GATE2,EFLAG,
BOCNT,OMCNT,X1STSG,ABSMX,OMCT2,ETPFG,TIMJ,
CSUM,FUDFT,ASUBX,DSUBX,XLSUBX,ESUBX,SSUBX,
ZSUBX,XLSUXH,ADOTU,ADDEL,ASUXT,DSUXT,FACTR,OTYPE,
XMNUTS,XLX,DELTX,A2OR,UMUO,RSUBA,ROVA,
USUBA, DENOM, SS, OBRA1, OBRAN, TMM, SATEL, STAID,
XLAMBA, ALPHA, DELTA, ITIMES, RANGE,
Q, DRGCO, XKAE, R2PM1, XK, EMIS,
SIDRT,THGRO,FLP25,VCO3,SIGS,SQRMU,XMU,
THDOT,F,XKE,RHOO,CDO,XJMPRM,XKMPRM,
SIXP9,HMPRM,ONEPI,RADYN,XMPER,LINE,
DEN7, FLAG4, VFLAG, XMASS, TS, DC, VDATA, ISTORV,
RDDTU,RDOTA,UDOTA,RODTO,RHODT,EXOM,EYOM,RCUBE,
XMUA32,ABMX2,GATE1,GATE2,XJAY5,SUM2,RCNT2,
LSTCNT,SPHA,CPHA,SDEL,CCDEL,
ADEL,BDEL,CDEL,ABDEL,FDEL,HDELA,HDELB,HDELC,HDELX,CAPXX,
ARAY,XTVEC,YTVEC,ZTVEC,BXSS,BYSS,BZSS,BCOF,BAB,BIG,BBUF,CVERT,
AVEC, XVEC, YVEC, ZVEC,ASIG1, IAXD, IBMUTH,
ISTORC,DLXV,RHOV,ADDLV,C,DC,IMOVOB$
DIMENSION Q(2),AXN(2),HX(3),FRAC(2),WX(3),AX(3),A(2),XMX(3),
XNX(3),AXNI(2),XNODE(2),UO(2),UX(3),VX(3),X(3),XDOT(3),
 XNUX(3), EQX(3) ,
CNTWD(27),STACD(6),OBSCD(6),SINPH(2),CAPX(3),RHOX(3),TXYZ(4),
TERMS(25),WSTO(25),TIMTB(6),CSUBI(6),X1STSG(9),ASUBX(3),
 DSUBX(3), XLSUBX(3), OTPT(8), DLTAA(6),
ESUBX(3),SSUBX(3),ZSUBX(3),XLSUXH(3),ADOTU(3),ASUXT(3),
DSUXT(3),XLX(3),DELTX(3),RSUBA(4),USUBA(4),
 RKSTO(79), HXI(3),
RKCNT(90), VDATA(1000), DLXV(3),
```

```
         FJOFT(7),ARAY(3),XTVEC(6),YTVEC(6),ZTVEC(6),
         RDOTU(3),RDOTA(4),UDOTA(4),RHODT(4),
         BXSS(6),BYSS(6),BZSS(4),BCOF(20),BAB(3),
         BIG(6),OBUF(26),CVERT(36),AVEC(3),XVEC(6),
         YVEC(6),ZVEC(6), ASIG1(4),CAPXX(3), CAPD(3),
         ADEL(3),BDEL(3),CDEL(3),ABDEL(3),FDEL(3),
         HDELA(2),HDELB(2), HDELC(2),HDELX(3)$
         DIMENSION QLX(3) $
*
         ADOTU=0.0@ADOTV=0.0@ADOTW=0.0@ADDEL=0.0$
         DO (B1) I = 1,25 $
 B1      TERMS(I)=0.0$
         DO 1 I=1,3$
         ADOTU=ADOTU+UX(I)*ARAY(I)$
         ADOTV=ADOTV+VX(I)*ARAY(I)$
         ADOTW=ADOTW+WX(I)*ARAY(I)$
    1     ADDEL = ADDEL + DELTX(I) * ARAY(I)$
         BX=X(1)+CAPX(1)@BY=X(2)+CAPX(2)@BZ=X(3)+CAPX(3)$
         BXX=BX**2@BYY=BY**2@BZZ=BZ**2                $
         BRHO=SQRT(BXX+BYY+BZZ)                       $
         IF(IWSTO-3)A1,CORRD,A1$
    A1   I = 1 $        RANGE OR ANGLE OBSERVATION
 CDLTA   TERMS(I)=ADOTU*RSUBA+ADOTV*USUBA$
         I=I+1$
         DLTAA=TERMS(I-1)$
        3TERMS(I)=ADOTV*UXN+ADOTU*RXN$
         DLTAX=TERMS(I)@I=I+1$
        5TERMS(I)=ADOTU*RYN+ADOTV*UYN$
         DLTAY=TERMS(I)@I=I+1$
        7TERMS(I)=ADOTV*USUBU+ADOTU*RSUBU$
         DLTLO=TERMS(I)@I=I+1$
        9TERMS(I)=ADOTV*WZ*R+ADOTW*R*(-XMZ)*COSU$
         DLTMD=TERMS(I)@I=I+1$
       11TERMS(I)=R*SINU*ADOTW$
         DLTIN=TERMS(I)@I=I+1$
 12       IF (IWSTO) 14,13,14$   FIND OBSERVATION TYPE
*      RANGE OBSERVATION
   13    TERMS (LSTCNT+1) = WSTO(16) - RHOC@RESID = TERMS(LSTCNT+1)*
         XMPER   $
          DO 60  K = 1, 6 $
 60       TERMS(K)=TERMS(K)/ASIG1 $  APPLY WEIGHT TO EQUATION
         TERMS(LSTCNT+1)=TERMS(LSTCNT+1)/ASIG1(1) $
         CALL SETI(IAXD,I)$
         BAX=BX/BRHO@BAY=BY/BRHO@BAZ=BZ/BRHO          $
         BAXX=COSTH*BAX+SINTH*BAY  $
         BAYY=-SINTH*BAX+COSTH*BAY  $
         BAX=BAXX  $
         BAY=BAYY  $
         TERMS(I)=BAX/ASIG1                           $
         I=I+1                                        $
         TERMS(I)=BAY/ASIG1                           $
         I=I+1                                        $
         TERMS(I)=BAZ/ASIG1                           $
         GO TO TSTOBS$
* ANGLE OBSERVATION
 14      TERMS(LSTCNT+1)=ADDEL*RHOC@RESID=TERMS(LSTCNT+1)*XMPER  $
         IF(SENSE LIGHT3)90,20$
* DECL OR ELEV PASS
 90      DON=ASIG1(3)*BRHO $
         SENSE LIGHT 3  $
         GO TO 22        $
```

```
   80  DO. =  SIG1(4) * RH-C $
 82     DO 1 . =1,6     $
  81   TE. S(. ) = TER S(K) / .CN $
       TE. S (LSTC T+ ) = TER S (LSTCNT+1) / DON $
       CALL SETI(I XD,I)$
       TER S(1)=(COSTH*ARAY(1)+SINTH*ARAY(2))/DON$
       I=I+1  $
       TER S(I)=(-SINTH*ARAY(1)+COSTH*ARAY(2))/DON$
       I=I+1  $
       TER S(I)=ARAY(3)/DON    $
       GO TO TSTOBS$
*
*    RANGE-RATE OBSERVATION
 CORRD  RHODT=0.0$
       DO20 I=1,3$
       RHODT(I+1)=XDOT(I)+WSTO(I+7)$
   20   RHODT = ARAY(I) * RHODT(I+1) + RHODT$
       EXOM=A*(ECOSE+ESQ-1)$
       EYOM=A*RTESQ*ESINE$
       VDOT=RVDOT/R$
       RSQR=R*R$
       RCUBE=RSQR*R$
       XMUA32=SQRMU*A*(SQRT(A(1)))$
       BARF=XMUA32*A/RCUBE$
       BLECH=1.0+RSQR/(A*P)$
       RDOTU=XMUA32*EXOM/RCUBE$
       RDOTA=-RDOT/2.0+UMUO*RDOTU$
       RDTXN=BARF*(SINEO-AXNI*ESINE-AYNI)$
       RDTYN=BARF*(-COSEO-AYNI*ESINE+AXNI)$
       UDOTU=-1.0*XMUA32*EYOM/RCUBE$
       UDOTA=-RVDOT/2.0+UMUO*UDOTU$
       UDTXN=BARF*RTESQ*(COSEO-AXNI*BLECH)$
       UDTYN=BARF*RTESQ*(SINEO-AYNI*BLECH)$
       DO 21 I=1,3$
  21     RDDTU(I) = 0.0$
       DO 22 I=1,3$
       RDDTU=RHODT(I+1)*UX(I)+RDDTU$
       RDDTV=RHODT(I+1)*VX(I)+RDDTV$
  22     RDDTW = RHODT(I+1) * WX(I) + RDDTW$
       I=1$J=1$
 CLOOP24PUC=RHOC*(RDOTA(J)-VDOT*USUBA(J))-RHODT*RSUBA(J)$
       GLUG=RHOC*(UDOTA(J)+(RDOT*USUBA(J))/R)-RHODT*USUBA(J)$
       TER S(I)=ADOTU*PUC+RDDTU*RSUBA(J)+ADOTV*GLUG+RDDTV*USUBA(J)$
       I=I+1$
     23J=J+1$
       IF(J-5)CLOOP24,UGH  25,UGH  25 $
 UGH  25GLUG=-1.0*ADOTU*RHOC*RVDOT*COSI+ADOTV*COSI*(RHOC*RDOT
       -RHODT*R)+RDDTV*R*COSI$
       TER S(I)=GLUG+ADOTW*SINI*(RHOC*(RVDOT*SINU-RDOT*COSU)
       +RHODT*R*COSU)-RDDTW*R*SINI*COSU$
       I=I+1$
     26J=J+1$
     27TER S(I)=ADOTW*(RHOC*(RVDOT*COSU+RDOT*SINU)-RHODT*R*SINU)
       +RDDTW*R*SI U$
       I=I+1$
     28RESID=WSTO(16)-RHODT$
       DON=ASIG1(2)*RHOC*3792243./108.E8 $
       TERMS(LSTCNT+1)=RESID*RHOC/DON  $
       RESID = RESID * 7.905 $
       DO 70 K = 1, 6 $
   70   TERMS(K) = TERMS(K) / DON $
```

```
          CALL SETI(IAXD,I)$
          BAX=XDOT(1)+0.05883447*X(2)$
          BAY=XDOT(2)-0.05883447*X(1)$
          BAZ=XDOT(3) $
          BAXX=COSTH*BAX+SINTH*BAY  $
          BAYY=-SINTH*BAX+COSTH*BAY  $
          BAX=BAXX  $
          BAY=BAYY  $
          TERMS(I)=BAX/DOM $
          I=I+1                                           $
          TERMS(I)=BAY/DOM $
          I=I+1                                           $
          TERMS(I)=BAZ/DOM $
 TSTOBS   IF (IWSTO - 2) AGATE, AGATE, BGATE $
 AGATE    IF(ABS(RESID)-ABSMX)16,BADOB,BADOB$
       16IF(ABS(RESID)-GATE1)17,17,OMIT1$
       17NTRAP=1$
       18RETURN$
 BGATE    IF(ABS(RESID)-ABMX2)B16,BADOB,BADOB$
      B16IF(ABS(RESID)-GATE2)17,17,OMIT1$
 BADOB    NTRAP=2$
          GO TO 18$
 OMIT1    NTRAP=3$
          GO TO 18$
          START TAC$
P PI      F/3.141592653 $
P PIOV2   F/1.570796326 $
P CONV    F/.01745329251 $      DEGREES TO RADIANS
P TWOPI   F/6.283185306 $
          SYMBOUT COEFF.TSTOBS$
          PAGE$
          END TAC$
          END$
          SUBROUTINE GTOBS (N2)$
*         ENTER DIMENSION, COMMON, AND EQIV. HERE$
          EQUIVALENCE(RKSTO,RKCNT(2)),(HYI,RKCNT(11)),(HZ I,RKCNT(12)),
                     (HXI,RKCNT(10)),(AXI,RKCNT(7)),(AYI,RKCNT(8)),
          (AZI,RKCNT(9)),(AYN,AXN(2)),(HY,HX(2)),(HZ,HX(3)),
          (WY,WX(2)),(WZ,WX(3)),(AY,AX(2)),(AZ,AX(3)),
          (XMY,XMX(2)),(XMZ,XMX(3)),(XNY,XNX(2)),(XNZ,XNX(3)),
          (AYNI,AXNI(2)),(UY,UX(2)),(UZ,UX(3)),(VY,VX(2)),(VZ,VX(3)),
          (Y,X(2)),(Z,X(3)),(YDOT,XDOT(2)),(ZDOT,XDOT(3)),
          (XNUY,XNUX(2)), (XNUZ, XNUX(3)),(EQY  , EQX(2)) ,
          (EQZ, EQX(3)), (COSPH, SINPH(2)), (CAPY, CAPX(2)),
          (CAPZ,CAPX(3)),(RHOY,RHOX(2)),(RHOZ,RHOX(3)),
          (X2NDSG,X1STSG(2)),(X3RDSG,X1STSG(3)),(X4THSG,X1STSG(4)),
          (X5THSG,X1STSG(5)),(X6THSG,X1STSG(6)),(X7THSG,X1STSG(7)),
          (X8THSG,X1STSG(8)),(X9THSG,X1STSG(9)),(ASUBY,ASUBX(2)),
          (ASUBZ,ASUBX(3)),(DSUBY,DSUBX(2)),(DSUBZ,DSUBX(3)),
          (XLSUBY,XLSUBX(2)),(XLSUBZ,XLSUBX(3)),(ESUBY,ESUBX(2)),
          (ESUBZ,ESUBX(3)),(SSUBY,SSUBX(2)),(SSUBZ,SSUBX(3)),
          (ZSUBY,ZSUBX(2)),(ZSUBZ,ZSUBX(3)),(XLSUYH,XLSUXH(2)),
          (XLSUZH,XLSUXH(3)),(ADOTV,ADOTU(2)),(ADOTW,ADOTU(3)),
          (ASUYT,ASUXT(2)),(ASUZT,ASUXT(3)),(DSUYT,DSUXT(2)),
          (DSUZT,DSUXT(3)),(XLY,XLX(2)),(XLZ,XLX(3)),(DELTY,DELTX(2)),
          (DELTZ, DELTX(3)), (IRPT, CNTWD(4)), (IWSTO, WSTO(3)) ,
          (DLYV,  DLXV(2)), (  DLZV, DLXV(3)) ,
          (RXN,RSUBA(2)),(RYN,RSUBA(3)),(RSUBU,RSUBA(4)),(UXN,USUBA(2)),
          (UYN,USUBA(3)),(USUBU,USUBA(4)),(RDDTV,RDDTU(2)),(RDDTW,
          RDDTU(3)),(RDTXN,RDOTA(2)),(RDTYN,RDOTA(3)),(RDOTV,
          RDOTA(4)),(UDTXN,UDOTA(2)),(UDTYN,UDOTA(3)),(UDOTU,UDOTA(4)),
```

```
(R DTU, UOTA(4)),(IAXD,STARD),(IAXE,STARE),
(D  ,C  D(2)),(DDG ,CARD(3)) $
EQUIVALENCE(DLTAA(2),DLTAX),(DLTAA(3),DLTAY),(DLTAA(4),DLTLO),
 (DLTA (5), DLT D), (DLTAA(6), DLTIN) $
CO  ON TO,  T, TF,   ,  V, THGR, XLO, AXN,
HX, X , RGC T,COUNT,FRAC,P,RTP,WX,SINI,COSI,SINO,
COSO,AX,CO  ,CUNTS,ESQ,E,A,XMX,XNX,AXNI,
XNODE, ,UO,EO1,EO2,SINEO,COSEO,ESINE,ECOSE,R,
SI  ,COSU,A ,RTESQ,UX,VX,X,XDOT,RDOT,RTA,RVDOT,
XNUX,  U , UZETH, RHO, XNUSQR, TS, C, DC, XOVR5, ZOVR5,
 RSQR, RRDOT, EGX, YLAT, XLONG, RKCNT,
OTPT,C TWD,STACD,OBSCD,SINPH,CAPD,CAPC,CAPS,
CAPX,OALT,PHIRD,XOVCT,SINTH,COSTH,THTA,TXYZ,
RHOX,RHO1,SMLH,TERMS,WSTO,TIMTR,CSUBI,XLMPTH,XINCL,DLTAA,
RHOC,SU ,ARSDL,XLGR,
RCNT,BSCNT,RESID,EFCNT,LSTAD,GATE2,EFLAG,
BOCNT,G CNT,X1STSG,ABSMX,OMCT2,ETPFG,TIMJ,
CSUM,FUDFT,ASUBX,DSUBX,XLSUBX,ESUBX,SSUBX,
ZSUBX,XLSUXH,ADOTU,ADDEL,ASUXT,DSUXT,FACTR,OTYPE,
XMNTS, LX,DELTX,A2OR,UMUO,RSUBA,ROVA,
USUBA,  ENO  , SS, OBRA1, OBRAN, TMM, SATEL, STAID,
XLAMBA, ALPHA, DELTA, ITIMES, RANGE,
Q, RGC , XKAE, R2PM1, XK, EMIS,
SIFT,THGRO,FLP25,VCO3,SIGS,SQRMU,XMU,
THDOT,F,XKE,RHOO,CDO,XJMPRM,XKMPRM,
SIXPQ, PR ,ONEPI,RADYN,XMPER,LINE,
DEN7, FLAG4, VFLAG, XMASS, TS, DC, VDATA, ISTORV,
RDOTU,RDOTA,UDOTA,RODTU,RHODT,EXOM,EYOM,RCUBE,
XMUA32,ABMX2,GATE1,GATE2,XJAY5,SUM2,RCNT2,
LSTCNT,SPHA,CPHA,SDEL,CCDEL,
ADEL,BDEL,CDEL,ABDEL,FDEL,HDELA,HDELB,HDELC,HDELX,CAPXX,
ARAY,XTVEC,YTVEC,ZTVEC,BXSS,BYSS,BZSS,BCOF,BAB,BIG,BBUF,CVERT,
AVEC, XVEC, YVEC, ZVEC,ASIG1, IAXD, IBMUTH,
ISTORC,DLXV,RHOV,ADDLV,C,DC,IMOVOB$
DIMENSION Q(2),AXN(2),HX(3),FRAC(2),WX(3),AX(3),A(2),XMX(3),
XNX(3),AXNI(2),XNODE(2),UO(2),UX(3),VX(3),X(3),XDOT(3),
 XNUX(3), EGX(3) ,
CNTWD(27),STACD(6),OBSCD(6),SINPH(2),CAPX(3),RHOX(3),TXYZ(4),
TERMS(25),WSTO(25),TIMTR(6),CSUBI(6),X1STSG(9),ASUBX(3),
 DSUBX( ), XLSUBX(3), OTPT(8), DLTAA(6),
ESUBX(3),SSUBX(3),ZSUBX(3),XLSUXH(3),ADOTU(3),ASUXT(3),
DSUXT(3),XLX(3),DELTX(3),RSUBA(4),USUBA(4),
 RKSTO(79), HXI(3),
RKCNT(  ), VDATA(1000), DLXV(3),
FUDFT(7),ARAY(3),XTVEC(6),YTVEC(6),ZTVEC(6),
RDOTU(3),RDOTA(4),UDOTA(4),RHODT(4),
BXSS(6),BYSS(6),BZSS(4),BCOF(20),BAB(3),
BIG(6),BBUF(26),CVERT(36),AVEC(3),XVEC(6),
YVEC(6),ZVEC(6), ASIG1(4),CAPXX(3), CAPD(3),
ADEL(3),BDEL(3),CDEL(3),ABDEL(3),FDEL(3),
HDELA(2),HDELB(2), HDELC(2),HDELX(3)$
 IF(RANGE)3, ,3$
2IF(ALPH )4,5,4$
5IF(DELT )4,6,4$
6IF(RODT )RRATE,X6,RRATE$
CASE SE LIGHT 5$
 GO TO 
3DO 7 I= ,8$
7ASUBX(I)=0.0$
 ASUBX(9)=RANGE $
  N2=0              $
```

```
      GO TO
      DELTA= LTA*CO  $
      AL  A = LP  *C  V$
      IF(  P  )A  I   A
      AL  HA = -ALPHA.
      AS  X(1)= (- S  (ALPHA))
      SP  Z=AS  FY( )
      XL  BZ=     SI (DELTA)  $
      SDEL=XL UBZ
      ASUBY = COS (ALPHA)$
      CP  =A  BY
      DSUBY=  SU Z*A UBX$
      DSU Z=C S(DELT )$
      ASIG1( )=ASIG1(3)/DSUBZ $ TAKE COS OF DEC OUT OF RA SIGMA
      CODEL=  UBZ
      XLSUBX= SUBY*DSUBZ$
      AS  Z= .0 $
      DS  X=(-XLSUBZ)*ASUBY $
      XLSUBY= SUBZ*ASUBX*(-1.0) $
      N2= $
      GO TO
A   ASUYT=C S (ALP A)$
      CP  =A  YT
      AS  T=  N (ALP A)$
      SP  A=AS XT( )$
      XLSUZH= IN (DE TA)$
      SDEL=X  UZH
      DS  T=  S ( EL  )$
      CODEL=  UZT
      XL  XH=(DSUZT* SUYT)*(-1.0)     $
      XLSUYH=(ASU T* SUZT)$
      AS  T= .0$
      DS  T=  SUZ *(-ASUXT)$
      DSUXT=XLSUZH* SUYT     $
      ESUBX=(-SI TH) 
      ESUBY=  STH
      ES  Z= .0$
      ZS  Z = SI  H( )     $
      SSU Y = (-ZSUB )* ESUB      $
      SS  X= SUBZ*ES  Y$SSUBZ=(-C SPH)$
      ZSU Y= SUBZ*ES  X$ZSUBZ=(-SSUBZ)*ESUBY$
      DO 10  I=1,3     $
      XLSUBX(I)=XLSUY *ESUBX(I)+XLSUXH*SSUBX(I)+
      XL  ZH* SU  (I)    $
      AS  X(I)=ASUXT*SSUBX(I)+ASUYT*ESUBX(I)+ASUZT*ZSUBX(I)$
      DSUBX( ) = DS XT*SSUBX(I)+DSUYT*ESUBX(I)+DSUZT*ZSUBX(I)$
      N2=1$
      GO TO
  IF  DO 11  I=1,
  1    A  X( )=0.0$
      AS  X=-(CA  *0. 58 7447)$
      AS  X( )=C   *  .05  3447$
      AS  X( )=P  TO
      N2= $
      RE  
      ST  T   0$
P     EZ. 17  32  51 $     DEGREES TO RADIANS
      FA      $
      EN  TA  $
      
      SU       E  LP  (I,IT,SLAT,SLO ,RS, R, LA,DEC)
```

```
      CALL MONEFF(T,TT,SMLAT,SMLON,RA,DEC)*
      U1=COSF(SLAT)*COSF(SLON)                                       $
      U2=COSF(SLAT)*SINF(SLON)                                       $
      U3=SINF(SLAT)                                                  $
      V1=COSF(SMLAT)*COSF(SMLON)                                     $
      V2=COSF(SMLAT)*SINF(SMLON)                                     $
      V3=SINF(SMLAT)                                                 $
      S=U1*V1+U2*V2+U3*V3                                            $
      XM=.0123115578                                                 $
      RM=60.33636791                                                 $
      DR=XM/RM *(2.*RS)/(RM*RM )*(1.5*S*S-.5)                        $
      DSLA=-SINF(SLAT)*(COSF(SLON)*V1+SINF(SLON)*V2)+COSF(SLAT)*V3   $
      DSLO=-SINF(SLON)*COSF(SLAT)*V1+COSF(SLON)*COSF(SLAT)*V2        $
      DLA=XM/RM * (RS*RS)/(RM*RM) * 3.* S * DSLA                     $
      DLO=XM/RM * (RS*RS)/(RM*RM) * 3.* S * DSLO                     $
      RETURN                                                         $
      END (                                                          $
      SUBROUTINE MONEF(IYR,DAY,CLAT,CLON,R,DEC)                      $
      RAD=57.2957795                                                 $
      SEC=206265.                                                    $
      TP=6.283185307                                                 $
      PI=3.141592653                                                 $
      IF(IYR-6)30,32,34                                              $
   30 T=DAY+731.5                                                    $
      GO TO 115                                                      $
   32 T=DAY+1096.5                                                   $
      GO TO 115                                                      $
   34 T=DAY+1461.5                                                   $
  115 XL=MODF(5.200850+0.22997150*T,TP)                              $
      FL=MODF(0.739177+0.22802713*T,TP)                              $
      FLP=MODF(6.229427+0.01720197*T,TP)                             $
      FX=MODF(2.031039+0.23089572*T,TP)                              $
      D=MODF(0.328366+0.21276872*T,TP)                               $
      E=MODF(D+D,TP)                                                 $
      G=MODF(FL+FL,TP)                                               $
      H=MODF(FX+FX,TP)                                               $
      CLON =    SEC*XL+22640.*SINF(FL)-4586.*SINF(FL-E)+2370.*SINF(E)
      +769.*SINF(G)-668.*SINF(FLP)-412.*SINF(H)-212.*SINF(G-E)
      -205.*SINF(FL+FLP-E)+192.*SINF(FL+E)-165.*SINF(FLP-E)
      +148.*SINF(FL-FLP)-125.*SINF(D)-109.*SINF(FL+FLP)-55.*SINF(H-E)
      -45.*SINF(FL+H)+40.*SINF(FL-H)-38.*SINF(FL-E-E)+36.*SINF(FL+G)
      -31.*SINF(G-E-E)+28.*SINF(FL-FLP-E)-24.*SINF(FLP+E)
      +19.*SINF(FL-D)+18.*SINF(FLP+D)                                $
      CLON=MODF(CLON,1296000.)/SEC                                   $
      S=FX+.1096*SINF(FL)-.0222*SINF(FL-E)+.0115*SINF(E)
      +.0037*SINF(G) $
      CLAT=18520.*SINF(S)*(1.-.00293*SINF(1.403808-.0009242203*T))
      -31.*SINF(FX-FL-E)-25.*SINF(FX-G)-23.*SINF(FLP+FX-E)
      +21.*SINF(FX-FL)-526.*SINF(FX-E)+44.*SINF(FL+FX-E)             $
      CLAT=CLAT/SEC                                                  $
      CA=COSF(CLON)                                                  $
      CB=COSF(CLAT)                                                  $
      SA=SINF(CLON)                                                  $
      SB=SINF(CLAT)                                                  $
      SD=.91745*SB+.39784*CB*SA                                      $
      CDCA=CB*CA                                                     $
      CDSA=-.39784*SB+.91745*CB*SA                                   $
      IF(ABSF(CDCA)-ABSF(CDSA))3,3,4                                 $
    3 R=ATANF(CDCA/CDSA)                                             $
      R=1.570796-R                                                   $
      GO TO 6                                                        $
```

```
 4    R=ATANF(CDSA/CDCA)                                          $
      IF(R)5,6,6                                                  $
 5    R=R+PI                                                      $
 6    IF(CDS-)7,8,8                                               $
 7    R=R+PI                                                      $
 8    DEC=ATANF(SD/SQRTF(1.-SD*SD))                               $
 99   RETURN                                                      $
      END (                                                       $
      SUBROUTINE PRESS(IXYR,X,T,D,XM,VV,U1,U2,U3)$
      DIMENSION X(3)$
      TABLEDEF SUNL(4),C14(4),C15(4),C16(4),C17(4) $
      C1= .9856472             $
      P1=.456E-6   $
      P2=1.006814784           $
      PI=3.141592653           $
      HRK=0.0 $
      IF(IXYR-6)10,11,12       $
10    I=1      $
      GO TO 13$
11    I=2      $
      GO TO 13  $
   12 IF(IXYR-7)100,101,100 $
  101 I=3$
13    SOL=SUNL(I)+C1*T+C15(I)*SINF(C1*T-C14(I))   $
      RA=SOL-C16(I)*SINF(SOL+SOL) $
      DEC=ATANF(P2*C17(I)*SINF(RA))$
      CRA=COSF(RA)@CDEC=COSF(DEC)$
      SRA=SINF(RA)@SDEC=SINF(DEC)$
      U1=CRA*CDEC@U2=SRA*CDEC@U3=CDEC  $
      R1=SQRTF(X(1)*X(1)+X(2)*X(2)+X(3)*X(3))$
      V1=X(1)/R1@V2=X(2)/R1@V3=X(3)/R1$
      TT=U1*V1+U2*V2+U3*V3 $
      IF(TT)1,1,2  $
1     IF(ABSF(TT)-SQRTF(R1*R1-1.)/R1)2,3,3  $
2     V=1.0   $
      GO TO 4$
3     V=0.0   $
4     XSECT=D*D*PI/4.0$
      VV=-XSECT/XM*P1*V*HRK$
  100 RETURN  $
      START TAC $
SUNL  F/4.870848558 $
      F/4.866542762 $
      F/4.8621843  $ SUNL FOR 1964
      F/0.0          $
      F/0.0          $
C14   F/.0559°7282  $
      F/.054833582 $
      F/.0652514   $ C14 FOR 1964
      F/0.0          $
      F/0.0          $
C15   F/.0334502     $
      F/.0334514     $
      F/.0334484    $  C15  FOR 1964
      F/0.0          $
      F/0.0          $
C16   F/.043053055   $
      F/.043 52683   $
      F/.043 5219   $ C16 FOR 1964
      F/0.0          $
      F/0.0          $
```

```
4      R=ATANF(CDSA/CDCA)                                  $
       IF(R)5,6,6                                          $
5      R=R+PI                                              $
6      IF(CDS-)7,8,8                                       $
7      R=R+PI                                              $
8      DEC=ATANF(SD/SQRTF(1.-SD*SD))                       $
99     RETURN                                              $
       END (                                               $
       SUBROUTINE PRESS(IXYR,X,T,D,XM,VV,U1,U2,U3)$
       DIMENSION X(3)$
       TABLEDEF SUNL(4),C14(4),C15(4),C16(4),C17(4) $
       C1= .9856472            $
       P1=.456E-6   $
       P2=1.006814784          $
       PI=3.141592653          $
       HRK=0.0 $
       IF(IXYR-6)10,11,12      $
10     I=1     $
       GO TO 13$
11     I=2     $
       GO TO 13  $
   12  IF(IXYR-7)100,101,100 $
  101  I=3$
13     SOL=SUNL(I)+C1*T+C15(I)*SINF(C1*T-C14(I))  $
       RA=SOL-C16(I)*SINF(SOL+SOL) $
       DEC=ATANF(P2*C17(I)*SINF(RA))$
       CRA=COSF(RA)@CDEC=COSF(DEC)$
       SRA=SINF(RA)@SDEC=SINF(DEC)$
       U1=CRA*CDEC@U2=SRA*CDEC@U3=CDEC  $
       R1=SQRTF(X(1)*X(1)+X(2)*X(2)+X(3)*X(3))$
       V1=X(1)/R1@V2=X(2)/R1@V3=X(3)/R1$
       TT=U1*V1+U2*V2+U3*V3 $
       IF(TT)1,1,2  $
1      IF(ABSF(TT)-SQRTF(R1*R1-1.)/R1)2,3,3  $
2      V=1.0  $
       GO TO 4$
3      V=0.0  $
4      XSECT=D*D*PI/4.0$
       VV=-XSECT/XM*P1*V*HRK$
  100  RETURN  $
       START TAC $
SUNL   F/4.870848558 $
       F/4.866542762 $
       F/4.8621843  $ SUNL FOR 1964
       F/0.0         $
       F/0.0         $
C14    F/.055997282  $
       F/.05483335?2 $
       F/.0652514   $ C14 FOR 1964
       F/0.0         $
       F/0.0         $
C15    F/.0334502    $
       F/.0334?14    $
       F/.0334484   $  C15  FOR 1964
       F/0.0         $
       F/0.0         $
C16    F/.043?53055  $
       F/.043?5268?  $
       F/.043?219  $ C16 FOR 1964
       F/0.0         $
       F/0.0         $
```

```
C17     F/.43365539    $
        F/.43362265    $
        F/.43364999  $ C17 FOR 1964
        F/0.0    $
        F/0.0    $
        END TAC $
        END      $
        SUBROUTINE SUM(I,N,M,A,B)$
        DIMENSION I(18),B(10)$
        TABLEDEF ACNTS(23),QCNTS(53)                  $
        DO 1 L=1,10$
        IF(I(L))3,3,2$
2       TEMP=0.0$
        K=I(L)$
        DO 4 J=1,K$
        TEMP=A*TEMP+ACNTS(N)$
   4     N = N + 1   $
        B(M)=TEMP$
    1     M = M + 1 $
   3      RETURN $
        START TAC                 $
ACNTS   F/.160217285$
        F/.214584375$
        F/.328125$
        F/.75$
        F/1.0$
        F/.341796875$
        F/.46875$
        F/.75$
        F/2.0$
        F/.256347656$
        F/.322265625$
        F/.4375$
        F/.75$
        F/.205078125$
        F/.234375$
        F/.25$
        F/.128173828$
        F/.12890625$
        F/.109375$
        F/.068359375$
        F/.046875$
        F/.036621093 8$
        F/.021484375$
QCNTS   F/.074768066 4$
        F/-.854492188E-1$
        F/.09765625$
        F/-.1171875$
        F/.140625$
        F/-.1875$
        F/.25$
        F/-.5$
        F/1.0$
        F/-.134582520$
        F/.170898438$
        F/-.170898478$
        F/.234375$
        F/-.234375$
        F/.375$
        F/-.375$
        F/1.0$
```

```
          F/-1.0$
          F/.119628906$
          F/-.13671875$
          F/.146484375$
          F/-.17578125$
          F/.1875$
          F/-.25$
          F/.25$
          F/-.5$
          F/-.0897216797$
          F/.102539063$
          F/-.102539063$
          F/.1171875$
          F/-.1171875$
          F/.125$
          F/-.125$
          F/.0598144531$
          F/-.068359375$
          F/.05859375$
          F/-.0703125$
          F/.046875$
          F/-.0625$
          F/-.0384521484$
          F/.0341796875$
          F/-.0341796875$
          F/.0234375$
          F/-.0234375$
          F/.0170898438$
          F/-.01953125$
          F/.009765625$
          F/-.01171875$
          F/-.00961303711$
          F/.0048828125$
          F/-.0048828125$
          F/.00213623047$
          F/-.00244140625$
          ENDTAC               $
          END$
          SUBROUTINE RCARD (SATEL,STAID,ASIG1,IBMUTH,DAYNO,ALPHA,DELTA,
          RANGE,RODTO,UTIME,NCARD,XMPER)$
*
*   RCARD READS AND CONVERTS AN OBSERVATION CARD
*
          DIMENSION HARAY(7), D1(3), ASIG1(4) $
            TABLEDEF TMOTO (12) $
 READ     READ INPUT TAPE N,OCR,ENDER,SATEL,ITYPE,STAID,IY1,MONTH,DD,HH,
                XMM,SI,HARAY $
 OCR      FORMAT (A2,A3,I1,4X,W4,I1,I2,3F2.0,F6.4,A3,6A8) $
          ALPHA = 0.$ DELTA = 0.$ RODTO = 0.$ RANGE = 0. $
          DO 10 I = 1,4$
 10       ASIG1(I) = 0.0 $
            IF (ENDER) E (HEND) , NCARD = 2$ RETURN $
          NCARD = 1 $
          IF(ITYPE-4)RCRD2A,RCRD3A,RCRD2A $ JMP IF BASE FREQ
 RCRD2A   DAYNO = TMOTO(MONTH) + DD $
          ST  T  C$  CHECK FOR LEAP YEAR
          TMA       HARAY$
          SRA       30 $
          TA        HARAY
          INC L     IY1 $
          TM        IY1$
```

```
        SP        32$
        JOC       SW2          JMP IF NOT LEAP YEAR
        JOC       SW2          JMP IF NOT LEAP YEAR
        END TAC
         IF (MONTH) T( ), DAYNO=DAYNO+1.$    LEAP YEAR
  SW2   UTIME = 60. * HH + MIN + SI/60. $      GET OBS TIME IN MINUTES
        IF (IT. F - 2)  CR1, RCRD2, RCRD4 $
        START TAC $
        AFR D     40$
P READ  W/EI           $
*   RCRD3A READS IN EMITTED FREQUENCY,   JUMPS TO READ NEXT CARD
 RCRD3A TM        6/1T47   $
        ETA       HARAY+1 $
        TMW       HARAY+2  $
        JMP       FLTFLT.FXFLT
        TAW       EFREQ1
        JMP       READ $
        END TAC    $
*   OCR1 READS IN RANGE AND STD. DEV OF RANGE FOR RADAR OBSERVATION
 OCR1    IF(IRHD)HD2,HD1,HD2    $
 HD1     WRITE OUTPUT TAPE 5,HDD2 $
 HDD2    FORMAT(2H20,3HSTA,6X,4HTIME,5X,10HRANGE-E.R.,3X,5HSIGMA)$
         IRHD=10    $
         IOHD=0
         IDHD=0$
 HD2     CONTINUE   $
         START TAC $
         TMW      36/1T47   $
         ETA      HARAY + 1$
         TA       HARAY+1   $
         TMW      HARAY+2 $
         JMP      FLTFLT.FLTFLT
         FDA      XMPER$
         TO       RANGE$        RANGE IN EARTH RADII
         JMP      INDEX$     GET L(ASIG1) TO IR4
         CA       $
         TMW      HARAY+3$
         SLA      6 $
         SLA      6 $
         AM       0/33T47$
         SRA      36$
         JMP      FLTFLT.FXFLT   $
         FDA      XMPER
         TA       ,4 $
         END TAC    $
         WRITE OUTPUT TAPE 5,ROB1,STAID,UTIME,RANGE,ASIG1(1)$
 ROB1    FORMAT(2H20,I4,3X,F9.2,3X,E12.8,3X,F4.2)$
         START TAC$
         JMP      RTN $
 INDEX   TJ       (P)+6H4
         CD$
         TXDLC    0,4$
         TD       SAV1$
         TMC      ASIG1$
         TDXLC    0,4$
         JMP      2$
         END TAC $
 RCRD2   IF(IOHD)HD5,HD6,HD5 $
 HD6     WRITE OUTPUT TAPE 5,HDD4   $
 HDD4    FORMAT(2H20,3HSTA,6X,4HTIME,6X,9HALPHA-DEG,4X,9HDELTA-DEG,4X,
         9HSIG-ALPHA,3X,9HSIG-DELTA,3X,9H0=RA 1=AZ) $
```

```
         IOHD=100$
         IRHD=0 $
         IDHD=0 $
HE       CONTINUE$
         START DC $
         TMQ      HARAY+1$
         CA
         SLAQ     12                  0000003334
         TAM      D1+1                MINS
         CA
         SLAQ     12                  0000003536
         SLA      6
         AM       0/33T47             000003536.
         SRAQ     36                  0003536.3738 TO Q
         JMP      FLTFLT.FXFLT
         FDA      F/3600              CONVERT SECS TO DEGS OR HRS
         TQM      ALPHA
         TMQ      D1+1
         CA
         JMP      FLTFLT.FXFLT
         FDA      F/60                CONVERT MINS TO DEGS OR HRS
         TQA
         FAMS     ALPHA
         TMQ      HARAY$        30 - 32
         CA
         JMP      FLTFLT.FXFLT
         TAM      D2
         TMA      HARAY$    FIND WHETHER RA-DEC OR AZ-ALT
         SRA      12
         SM       0/40T47
         JAZ      (P)+6H
         TMA      D/1B15$
         TAM      IBMUTH
         TMA      D2
         FAM      ALPHA
         JMP      (P)+6H$
         CM       IBMUTH$      RA-DEC
         TMA      D2
         FSM      ALPHA
         TAQ
         FMQR     F/15                CONVERT AZIM FROM HRS TO DEGS
         TAM      ALPHA               -RT ASCEN IN DEGS TO ALPHA
         TMQ      0/7777T47
         ETA      HARAY+1$        0000003940 TO A
         TMQ      HARAY+2$        41-48 TO Q
         SLAQ     12                  000039-42 TO A,43-4800 TO Q
         TAM      D1                  DEGS
         CA
         SLAQ     12                  0000004344
         TAM      D1+1$         MINS
         CA
         SLAQ     12                  0000004546
         SLA      6
         AM       0/33T47
         SLAQ     6                   00004546.47 TO A
         TA
         CA
         JMP      FLTFLT.FXFLT
         FDA      F/3600              CONVERT SECS TO DEGS
         TQ       DELTA
         TM       D1+1
```

```
        CA
        JMP     FLTFLT.FXFLT
        FDA     F/6                     CONVERT TNS TO DEGS
        TCA
        FAMS    DELTA
        TMA     D1
        CA
        JMP     FLTFLT.FXFLT
        TAM     D1       $
        TMA     FLTFLT.MSIGN $
        JAZ     (P)+4H        $
        TMA     D1            $
        FSMS    DELTA   $
        JMP     (P)+3H  $
        TMA     D1      $
        FAMS    DELTA   $
        TMA     HARAY+4$
        SLA     12$        GET 0059-62
        SRAQ    24
        SRAQ    12
        SLA     6
        AM      0/33T47
        SRAQ    36
        JMP     FLTFLT.FXFLT
        CM      HARAY+6
        TAM     HARAY+6
        TMA     HARAY+6
        JMP     INDEX $
        TAM     2,4$
        TMQ     12/1T47$
        ETA     HARAY+4$
        TMQ     HARAY+5$
        SLA     6
        AM      0/33T47
        SRAQ    36
        JMP     FLTFLT.FXFLT
        TAM     3,4$
        END TAC $
        WRITE OUTPUT TAPE 5,OOB1,STAID,UTIME,ALPHA,DELTA,ASIG1(3),
        ASIG1(4),IBRUTH $
OOB1    FORMAT(2H20,W4,3X,F8.2,3X,F10.6,3X,F11.6,5X,F5.2,8X,F5.2,
         8X,I1) $
        DO(CV) I=3,4 $
CV      ASIG1(I)=ASIG1(I)*.000005 $  CONVERT SIGMA TO RADIANS
        START TAC $
        JMP      RTN $
TMOTO   F/0.0$
        F/31.0$
        F/59.0$
        F/90.0$
        F/120.0$
        F/151.0$
        F/181.0$
        F/212.0$
        F/243.0$
        F/273.0$
        F/304.0$
        F/334.0$
        F/365.0$
* OCR3 READS IN DOPPLER OBSERVATION (RANGE RATE)
        END TAC $
```

```
RCRD4   IF(IDHC)HD3,HD4,HD3 $
HD4     WRITE OUTPUT TAPE 5,HDD3 $
HDD3    FORMAT(2H20,3HSTA,6X,4HTIME,6X,10HFREQ - CPS,4X,5HSIGMA) $
        IDHC=1 $
        IOC=0 $
        IRHC=0 $
HD3     CONTINUE $
        START TAC$
        TMQ       6/1T47  $
        ETA       HARAY+1 $
        TMQ       HARAY+2 $
        JMP       FLTFLT.FXFLT
        TAM       D1
        FAM       F/100.EP $
        TAM       D1+1  $
        FDA       F/100.  $
        TQM       D2     $
        TMA       D1  $
        FSM       EFREQ1  $
        FDA       D1+1  $
        FMMR      F/-37922.43
        TAM       RODTO
        JMP       INDEX $
        AIXO      1,4$
 OCF12  CA    $
        TMQ       HARAY+3 $
        SLAQ      6
        SLA       6
        AM        0/33T47
        SRAQ      36
        JMP       FLTFLT.FXFLT
        TAM       ,4$
        END TAC $
        WRITE OUTPUT TAPE 5,DOB1,STAID,UTIME,D2,ASIG1(2)$
 DOB1   FORMAT(2H20,W4,3X,F8.2,3X,F12.2,3X,F4.2)$
        START TAC$
 RTN    TMD       SAV1 $          RESTORE INDEX REGISTER 4
        TDXLC     0,4 $
E       AFEND
        END TAC  $
        RETURN $
        END $
        SUBROUTINE RCTFY(XMPER,ESINE,ECOSE,A,WX,DRGCO,RKSTO,XN,XMZ,AYNI,
        UZ,R)  $
*
*     SUBROUTINE RCTFY APPLIES THE CORRECTIONS FOR ATMOSPHERIC DRAG
*         DURING THE INTEGRATION OF THE EPHEMERIS
*    IT IS CALLED ONCE PER REVOLUTION BY CNTRL, INTEGRATION CONTROL
*
        EQUIVALENCE (BESI1, BESIO(2) ) $
        DIMENSION RKSTO(79),BESIO(10),BESI1(1),WX(3),A(2),Q(2)$
        START TAC $
P PI    F/3.141592653 $
P XKE   F/.0743652845
        END TAC $
        ECCEN=ATAN (ESINE/ECOSE)$
        IF(ECCE )1,2,2$
 1      ECCEN=PI+ECCEN$
 2      XMEANA=ECCEN-ESINE$TYTPI=XMEANA/XN$
        ESQ=RKSTO(6)**2+RKSTO(7)**2+RKSTO(8)**2          $
        E=SQRT (ESQ)$
```

```
      PE-Fr=-(1)*(1. -F)--=P RGE$
      ZE =(X T*AY 1* )/E  Z=ZED/    $
      H=-TLH( /,A,)F-FR)  $
      CALL T-I-T(I,H,SIGM ,X-F,XKFRG,RATIC) $
      GO TO ( ,6),I     $
      CALL BESSEL(XK-RG,A,E,-KAE,PTKAE,PESIO,PESI1,EMKAE,XKAESQ,Q) $
      CALL A  RAG(ES ,E,DPGC ,RATIC,A,Q,BESI ,PERGE,ADRAG,ODRAG)$
      ADRAG=A(1)+ADRAG  $
      XNDRAG=:KE/SQRT(ADRAG**3)$
      RKSTO(5)=RKSTO(5)+XDRAG*TWTPI-XMEANA    $
      FPEE=(1.0-(PERGE+GDRAG)/ADRAG)/E$
       PDRAG= .0$
      DO 3 K = 6,8 $
      RKSTO(K)=FPEE*RKSTO(K)$
    3 PDRAG = PDRAG+RKSTO(K)**2$
      PDRAG=SQRT(ADRAG*(1.0-PDRAG))  $
      DO 4 I= ,11$
    4  RKSTO(I) = WX(I-8)*PDRAG$
     $RETURN$
      END$
      SUBROUTINE READER (STARC,ISTCNT,LSTCNT,IASIZE,IDBUF,XMPER)     $
       DIMENSION STARC(1000), IDBUF(10)$
      TABLEDEF ASIG(4)$
*
*     READ STATION CARDS FROM TAPE 10
*     STORE STATION INFO IN 9 WORD GROUPS IN STARC BUFFER
*    ASSIGN INTERNAL DATUM NUMBERS AND STORE EXTERNAL DATUM NUMBERS
*     IN IDBUF TABLE
         ISTC T = 0Q REWIND 10$
      WRITE OUTPUT TAPE 5,SC1  $
 SC1  FORMAT(10H70  INPUT DATA - STATION CARDS/)$
      WRITE OUTPUT TAPE 5,SC2  $
 SC2  FORMAT(2H20,3HSTA,3X,8HLATITUDE,4X,9HLONGITUDE,4X,9HELEVATION,
      4X,5HDATUM,12X,18HOBSERVATION SIGMAS)$
      WRITE OUTPUT TAPE 5,SC5$
 SC5  FORMAT(2H10,53X,30HRANGE   RRATE   RA-AZ   DEC-EL)$
 KN1  J = ISTCNT * 9 + 1$        READ NEXT STATION CARD
      K=J+8$
      READ INPUT TAPE 10, KN3, (STARC(I),I = J,K)$
 KN3  FORMAT( 4,F8.5,F9.5,F6.0,5X,I4,1X,F8.8,3F8.5)$
      IF (STARC(J)) E (HEND), GO TO TBL $
      WRITE OUTPUT TAPE 5,SC3,(STARC(I),I=J,K)  $
 SC3  FORMAT(2H10,A4,1X,F9.5,3X,F10.5,4X,F7.0,5X,I4,4X,4F8.5) $
      L=J+5$
      M=1$                           REPLACE SIGMAS THAT
      DO 20 I=L,K$                   WERE NOT GIVEN
      IF(STARC(I)) 20, 10, 20$       WITH STANDARD VALUES
   10 STARC(I) = ASIG(M)$
   20 M = M + 1 $
      STARC(J+1) = STARC(J+1) * .01745329251 $ CONVERT LAT AND LONG TO
      STARC(J+2) =PRLPR(-STARC(J+2)*.01745329251)$      RADIANS
      STARC(J+3)=STARC(J+3)/XMPER   $
      STARC(J+5)=STARC(J+5)/XMPER  $
       ISTCNT = ISTC T + 1$     FORM STATION COUNT IN ISTCNT
       GO TO  KN1$         GO GET NEXT STATION CARD
        START TAC$
P HEND  W/0000E D $
 FOUND  TXDLC    ,INCUR$ THIS EXTERNAL DATUM NO. ALREADY IN TABLE
        TDA      $ REPLACE STARC ENTRY W. INTERNAL DATUM NO.
        SM       IDBUF$
        AFEND    72$
```

```
         TAM      ,INSTARC
         JMP      NEXT
ASIG     F/10.0   $ METERS - RANGE SIGMA
         F/10.0   $ CYCLES PER SECOND
         F/0.00001 $ RADIANS
         F/0.00001 $ RADIANS
         SAVE     INSTARC,3 $
         SAVE     INCUR,.    $
 TRL     CD$
         TXDLC    ,2
         TXDRC    ,3
         TDM      SAVE
         TMA      ISTCNT
         SLA      3
         AM       ISTCNT
         TMQ      STARC$
         TDXLC    ,INSTARC
         AIXO     4,INSTARC$
         CD$
         TXDLC    0,INSTARC $
         AD$
         AM       C/HLT,0@C/JMP,TEST$
         TAM      DONEKEY$ C/HLT,STARC+4+9*ISTCNT@C/JMP,TEST$
         TMQ      28/1T27@8/1T47
         ETD      SEARCH
         TDM      SEARCH
TEST     TMA      ,INSTARC
         JAZ      NEXT$ IGNORE STARC ENTRY IF DATUM NO. ZERO
         TMD      IDBUF$
         TDXLC    0,INCUR$ IR = ADDR OF DATUM NO. TABLE
RSEARCH  RPTAN    0$ IS THIS DATUM NO. ALREADY IN TABLE<
         TMQ      1,INCUR
         JAED     FOUND$  YES -
NTFOUNDINCAR      SEARCH$  NO - FORM NEW INTERNAL DATUM NO.
         TMQ      0/7777T39
         ETD      SEARCH
         SCD      24$  REPLACE DATUM NO. IN STARC TABLE
         TDM      ,INSTARC$ W. INTERNAL DATUM NO.
         TAM      ,INCUR$ EXT. DATUM NO. TO DATUM NO. TABLE
NEXT     TMD      DONEKEY
         AIXO     9,INSTARC$
*                 GO TO **TEST** IF MORE DATA IN STARC ARRAY
FINI     TMQ      12/1T35$ END OF STARC TABLE
         ETD      SEARCH$ STORE SIZE OF DATUM NO. TABLE
         SCD      24$       IN IASIZE
         TDM      IASIZE$
         TDA
         SLA      1
         AD
         AM       D/6-15
         TAM      LSTCNT $      3 * IASIZE + 6
         TMD      SAVE
         TDXLC    ,2
         TDXRC    ,3
E        AFEND      $
         END TAC      $
 RTN     RETURN   $
          END $
         SUBROUTINE READIT(CDTAD,STARC,IDFD, CARD)$
          DIMENSION STARC (1000) , CDTAD (27) $
         READ INPUT TAPE 10,10,CDTAD(6), STARC(3),STARC(4),STARC(5) $
```

```
1·      FO· AT( 4,F  .5,F9.·,F6.·)$
        .C  I· = 1 -
        IF( ·T   (?))F( ···)·GO TO ·   $
        IF('FE ·50,·· ,  · ·
5.      WR··E ·  TPU  T ·E ·,SC 1 $
S··1    FO· AT( ···      PUT DA·A - STATION CARDS/)$
        WR··E ·  TPUT T ·E ·,SC·2 $
S··2    FO· AT( F2·,3F·TA,3·,8·LATITUDE,4X,9HLONGITUDE,4X,9HELEVATION/)$
        IP· =1· $
51      WRITE · TPUT T PE 5,SC·3, C TWD(8),STARC(3),STARC(4),STARC(5)$
SCC3    FO· AT(·H10,A4,2X,F·.5,4X,F9.5,5X,F6.0)$
        GO TO 3·$
3·      NCA·D=·$
31      RETURN$
         START TAC $
P HEND  W/END
        END TAC  $
        END  $
        SUBROUTINE STARTER (HD·G,CNTWD,X1STSG,TO,DT,TF,D,XM,XLO,AXN,HX,
        IP·TFL,·BSMX,MCODE,ORGLA,ORGTM,REFDA,REFTM,·BMX2,EFLAG,IXYR,
        XAXS)$
        DIMENSION X1STSG(9),HD·G(24),CNTWD(27),AXN(2),HX(3),MCODE(3)$
        DIMENSION CUT1(7), OUT2(6), OUT3(4) $
*
*       STARTER READS THE FIVE CONTROL CARDS
*       INPUTS O BITAL ELEMENTS AND CONTROL PARAMETERS FOR THE RUN
*
         N = 0
        READ INPUT TAPE N,CCR,(HDNG(I),I=1,10),(HDNG(I),I=11,20)$
 CCR    FORMAT(10A8) $
        READ INPUT TAPE N, CCR1, TO, DT, TF, D , XM $      CARD 3
 CCR1    FORMAT (6E12.·) $
        OUT3(1)=DT@OUT3(2)=TF  $
        OUT3(3)=D@OUT3(4)=X·   $
        READ INPUT TAPE N, CCR1, XLO, AXN, HX $      CARD 4
        IF(XLO)CCR5,CCR4,CCR4  $
 CCR5   XLO=-XLO            $
        OUT1(2)=XLO·OUT1(3)=AX (1)@OUT1(4)=AXN(2) $
        OUT1(5)=HX(1)@OUT1(6)=HX(2)@OUT1(7)=HX(3) $
        TEM1=SQRT(XLO/XAXS*(1.0-AXN(1)**2))$
        TEM2=(H (1)+HX(2)+HX(3))*.01745329       $
        TEM3=6.2831853·6  $
 CL3    IF(TEM2)GT(TEM3),GO TO CL1  $
        XLO=TEM2$
        GO TO CL21     $
 CL1    TEM2=TEM2-TEM3$
        GO TO C 3   $
 CL21   HX(2)=H (2)*.01745329    $
        HX(3)=H (3)*.01745329    $
        AX·(2)=·XN(2)*.01745329  $
 CL2    TE··  =·XN(1)*COS(HX(2)) $
        TE·5  = XN(·)*SIN(HX(2)) $
        TE·6  =TEM1*SI·(HX(3))*SIN(·XN(2))$
        TE·9=COS(HX(3)) $
        TE·8=-TEM9       $
        TE·7=T· 1*TEM9*SIN(·XN(2))   $
        TE·   =TEM1*COS(AXN(2)) $
        AX·(1)=·EM4 AX·(2)=TEM5 $
        HX(1)=TE·6·HX(2)=TE·7·HX(3)=TE·8 $
 CCR4   OUT2(1)=XLO·OUT2(2)=AX·(1)@OUT2(3)=AXN(2)$
        OUT2(4)=HX(1)@OUT2(5)=HX(2)@OUT2(6)=HX(3)$
```

```
      RE4  I.^UT TAP- N,CCR2,IPNTFL,ORGDA,ORGTM,C'TW?(1),CNT D(4),
      EFL/G,X'STSG(1),REFT A,EFTA,CODE,IXYR,ABSMX,(X1STSG(I),I=2,9),
      AP  2
COF2  FO  AT( 1,5 ,F ,0,F .0, 1,11,F1.0,F3.0,F4.0,F9.0,3I1,I2,F6.0,
      8F .0,F .0) $
      OUT1(1)=ORG A+ FGT /14 0.0     $
      WRITE  TPUT T FE 5, T AGE1    $
TPAGE1 FOR AT(2H70,28 ,44H RBIT PREDICTIO  AN  STATIO  LOCATO  PROGRAM/
      /)$
      IF(CNT  (1))E( LNK),GO TO TPAGE3        $
      WRITE  TPUT T PE 5, TPAGE2    $
TPAGE2 FOR AT(2H10,44X,14HSIMULATION RUN//)   $
      GO TO TPAGE      $
TPAGE3 WRITE  UTPUT TAPE 5, TPAGE4      $
TPAGE4 FOR AT(2H10,36X,27H IFFERENTIAL CORRECTION RUN//) $
TPAGE5 WRITE   TPUT TAPE 5,TPAGE6       $
TPAGE6 FO  AT(2H10,11H INPUT DATA//)   $
      WRITE  UTPUT TAPE 5, TPAGE7, OUT1(1), OUT2(1), OUT3(1) $
TPAGE7 FOR AT(2H10,9X,9HEPOCH DAY,F15.8,7X,3HL/0,F15.8,7X,7HDELTA T,
      F15.8/) $
      WRITE  UTPUT TAPE 5, TPAGE8, OUT1(2), OUT2(2), OUT3(2) $
TPAGE  FOR AT(2H10,9X,9HAXIS-  M ,F15.8,7X,3HAXN,F15.8,7X,7HT/0-T/F,
      F15.8/) $
      WRITE  UTPUT TAPE 5, TPAGE9, OUT1(3), OUT2(3) $
TPAGE9 FOR AT(2H10,9X,9HECCEN-TY ,F15.8,7X,3HAYN,F15.8/)$
      WRITE  UTPUT TAPE5, TPAGE10, OUT1(4), OUT2(4) $
TPAGE10FOR AT(2H10,9X,9HINCLI     ,F15.8,7X,3HHX  ,F15.8/)$
      WRITE  UTPUT TAPE 5, TPAGE11, OUT1(5), OUT2(5), OUT3(3) $
TPAGE11FOR AT(2H10,9X,9HMEAN ANOM,F15.8,7X,3HHY ,F15.8,7X,7HDIAM     ,
      F15.8/) $
      WRITE  UTPUT T PE 5, TPAGE12, OUT1(6), OUT2(6), OUT3(4) $
TPAGE12FOR AT(2H10,9X,9HARG PERG ,F15.8,7X,3HHZ  ,F15.8,7X,7HWEIGHT ,
      F15.8/) $
      WRITE  UTPUT TAPE5, TPAGE13, OUT1(7) $
TPAGE13FOR AT(2H10,9X,9HR A NODE ,F15.8)  $
      RETURN  $
      START TAC $
PBL K  W/0000000 $
      END TAC $
      END $
      FUNCTIO  CALH (UZ,R,XMPER)$
*     COMPUTES HEIGHT IN  ETERS ABOVE OBL TE EARTH
      UZSQR=U **2$
      CALH =  MPER *(F2*UZSQR-UZSQR**2*F1 - 1. + R )  $
      RETURN$
      START TAC$
*     EARTH FLATTENNI G CONSTANTS
P F1  F/.1685717364E-4 $   (3/2) * F**2
P F2  F/.3369387073 -2 $          F + (3/2) * F**2
      PAGE    $
      END TAC $
      EN $
      SUBROUTINE TRI T(IJ,H,SIGMA,XMF,XKPG, ATIO)$
      TABLEDEF HSONE(132)        $
      IJ=1$
      IF(H-HS E(130))1,1,6 $
      100 2 I= ,132,3$
      J=1$
      IF(H-HS E(J)) 3,3,2$
      2CONTINU $
      R TUR
```

```
  ?      IF (J- ) L'AL:, 5, 5$
L..LT    W-ITE  UTPUT  PE F, 1-, H $
 1        F. WAT (14 10.LTIT DE CF   ,E12.4) $
         STA T T C$
         TMA      C/-LT,,ABCD C/-LT,4$
         JMPL     6$
         RU  UT    $
         JMP      3$
LABCD    A/ALTIT DE  ELO. 50000 METEPS   $
         ENDTAC             $
        6H2MH1=HSGME(J)-HSGME(J-3)$
         H2MH=HSGME(J)-H$
         SIGMA=H2MH*HSGME(J-2)$
         HMH1=H-HSGME(J-3)$
         SIGMA=SIGMA+(HMH1)*(HSGME(J+1)) $
         SIGMA=SIGMA/H2MH1$
         XME=H2MH*HSGME(J-1)$
         XKORG=(HMH1*HSGME(J+2)+XME)/H2MH1$
         RATIO=EXP (SIGMA)$
         IJ=1     $
        6RETURN     $
         START TAC$
HSGME    F/50.0E3$
         F/-7.0310587$
         F/.13171E-02$
         F/55.0E3$
         F/-7.6032281$
         F/.12876E-2$
         F/60.0E3$
         F/-8.1538223$
         F/.118625E-02$
         F/65.0E3$
         F/-8.7536457$
         F/.108476E-02$
         F/70.0E3$
         F/-9.4124843$
         F/.98311899E-3$
         F/75.0E3$
         F/-10.103371$
         F/.8813 5E-3$
         F/80.0E3$
         F/-10.944509$
         F/.7795 E-3$
         F/85.0E3$
         F/-11.9 9229$
         F/.7806 E-3$
         F/90.0E3$
         F/-12.972557$
         F/.7818 E-3$
         F/95.0E3$
         F/-14. 71873$
         F/. 511 E-3
         F/100. -3$
         F/-15.  3575$
         F/. 44  -3$
         F/1 5.  3$
         F/-15. 11001$
         F/.1084 E-2
         F/110.  3$
         F/-16.8 36173$
         F/.1368 E-2$
```

F/115.0F3$
F/-17.625944$
F/.182[illegible]E-2$
F/120.0F3$
F/-18.2[illegible]1597$
F/.22922E-2$
F/125.0F3$
F/-18.7[illegible]5451$
F/.27547E-2$
F/130.0F3$
F/-19.1[illegible]2311$
F/.32187E-2$
F/135.0F3$
F/-19.5[illegible]3103$
F/.3682[illegible]E-2$
F/140.0F3$
F/-19.220952$
F/.41465E-2$
F/145.0F3$
F/-20.125377$
F/.46141E-2$
F/150.0F3$
F/-20.361461$
F/.50797E-2$
F/155.0F3$
F/-20.595472$
F/.5547E-2$
F/160.0F3$
F/-20.8[illegible]2337$
F/.60742E-2$
F/165.0F3$
F/-21.0[illegible]3537$
F/.64124E-2$
F/170.0F3$
F/-21.1[illegible]79[illegible]2$
F/.6682[illegible]E-2$
F/175.0F3$
F/-21.3[illegible]5866$
F/.69157E-2$
F/180.0F3$
F/-21.434534$
F/.70411E-2$
F/185.0F3$
F/-21.561085$
F/.7166[illegible]999E-2$
F/190.0F3$
F/-21.6[illegible]5493$
F/.7291[illegible]E-2$
F/195.[illegible]3$
F/-21.8[illegible]7622$
F/.74174E-2$
F/200.0F3$
F/-21.9[illegible]779[illegible]$
F/.75444E-2$
F/210.0F3$
F/-22.1[illegible]9273$
F/.7773[illegible]E-2$
F/220.[illegible]3$
F/-22.3[illegible]9072$
F/.79567E-2[illegible]
F/235.[illegible]3$

```
F/-22.6?934.$
F/.?23- E-2
F/?50.  3$
F/-?3.  ?85?$
F/.?57. E-2?
F/2?0.? 3$
F/-?3.4 614?$
F/.?88? ?-2$
F/?00.?
F/-?3.?7177?$
F/.?452?E-2?
F/330.0?3$
F/-?4.5 4617$
F/.10022E-1?
F/3?0.0?3$
F/-25.0?7847$
F/,1059?E-1?
F/400.0?3$
F/-25.6?742?$
F/.?137?-1$
F/450.0?3$
F/-26.?66557$
F/.1234?E-1?
F/500.0?3$
F/-27.??0063$
F/.1333?E-1?
F/?00.0?3$
F/-28.2?69017$
F/.?536?-1$
F/900.0?3$
F/-31.8?7479$
F/.2213?E-1$
PAGE   $
END TAC $
END  $
SUBROUTINE XEL(XEL1,XAXS,X1,X2,X3,X4,X5,X6)$
DIMENSION XEL1(7)$
XL=XEL1(1)$
A=XEL1(?) $
B=XEL1(?) $
C=XEL1(?) $
D=XEL1(?) $
E=XEL1(?) $
F=XEL1(7) $
RAD=.01745329252 $
P=D*D + E*E + F*F $
ES?=A*A + B*B + C*C $
AX=?/(1.-ES?) $
EX=SQRTF(ES?) $
WX=?/SQRTF(?) $
WY=E/SQRTF(?) $
WZ=?/SQRTF(?) $
XNZ=SQRTF(1.-WZ*WZ) $
XNX=-WY/XNZ $
XNY=XNX * ,? $
XNY=WX/?NZ ?
XNX=-X? * 7
XNZ=0. ?
AX =A * XNX +    *   Y
AY =A * XNX + ? * ? Y + C * X?Z $
X??=ATA?F(AY?/ X?) ?
```

```
 IF( AX)1,2,   
1XWO=XWO + 3.141592654
2IF(XO)3,4,  $
3XWO=XWO + 2.*3.141592654 $
4XWO=XWO/RA                $
 XNO=ABS (ASINF(XNY)) $
 IF(XNY)5,6,               $
5IF(XNX)7,8,               $
7XNO=XNO + 3.141592654     $
 GO TO 9                   $
8XNO = 2.*3.141592654 - XNO $
 GO TO 9                   $
6IF(XNX)10,9,9             $
10XNO=3.141592654 - XNO    $
9XNO=XNO/RAD               $
 X1=AX*AXS$
 X2=EX                     $ = ECCN
 X3=ASIN (XNZ)/RAD         $ = INC
 X4=XL/RAD - XNO -XWO      $ = MEAN ANOMALY
22IF(X4)20,21,21      $
20X4=X4+360.      $
 GO TO 22                  $
21IF(X4-360.)38,38,25  $
25X4=X4-360.  $
 GO TO 21  $
38X5=XWO                   $ = PERIGEE
 X6=XNO                    $ = NODE
 RETURN $
 END    $
 SUBROUTINE XYZSB$
XYZSB,  GIVEN A BAR, H BAR, AND XL AT SOME TIME T ------
 COMPUTES INTERMEDIATE ORBIT PARAMETERS AND SATELLITE POSITIONS AND
              VELOCITY VECTORS
 EQUIVALENCE(RKSTO,RKCNT(2)),(HYI,RKCNT(11)),(HZ I,RKCNT(12)),
             (HXI,RKCNT(10)),(AXI,RKCNT(7)),(AYI,RKCNT(8)),
 (AZI,RKCNT(9)),(AYN,AXN(2)),(HY,HX(2)),(HZ,HX(3)),
 (WY,WX(2)),(WZ,WX(3)),(AY,AX(2)),(AZ,AX(3)),
 (XWY,XWX(2)),(XWZ,XWX(3)),(XNY,XNX(2)),(XNZ,XNX(3)),
 (AYNI,AXNI(2)),(UY,UX(2)),(UZ,UX(3)),(VY,VX(2)),(VZ,VX(3)),
 (Y,X(2)),(Z,X(3)),(YDOT,XDOT(2)),(ZDOT,XDOT(3)),
 (XNUY,XNUX(2)), (XNUZ, XNUX(3)),(EQY  , EQX(2)) ,
 (EQZ, EQX(3)), (COSPH, SINPH(2)), (CAPY, CAPX(2)),
 (CAPZ,CAPX(3)),(RHOY,RHOX(2)),(RHOZ,RHOX(3)),
 (X2NDSG,X1STSG(2)),(X3RDSG,X1STSG(3)),(X4THSG,X1STSG(4)),
 (X5THSG,X1STSG(5)),(X6THSG,X1STSG(6)),(X7THSG,X1STSG(7)),
 (X8THSG,X1STSG(8)),(X9THSG,X1STSG(9)),(ASUBY,ASUBX(2)),
 (ASUBZ,ASUBX(3)),(DSUBY,DSUBX(2)),(DSUBZ,DSUBX(3)),
 (XLSUBY,XLSUBX(2)),(XLSUBZ,XLSUBX(3)),(ESUBY,ESUBX(2)),
 (ESUBZ,ESUBX(3)),(SSUBY,SSUBX(2)),(SSUBZ,SSUBX(3)),
 (ZSUBY,ZSUBX(2)),(ZSUBZ,ZSUBX(3)),(XLSUYH,XLSUXH(2)),
 (XLSUZH,XLSUXH(3)),(ADOTV,ADOTU(2)),(ADOTW,ADOTU(3)),
 (ASUYT,ASUXT(2)),(ASUZT,ASUXT(3)),(DSUYT,DSUXT(2)),
 (DSUZT, DSUXT(3)),(XLY,XLX(2)),(XLZ,XLX(3)),(DELTY,DELTX(2)),
 (DELTZ, DELTX(3)), (IRPT, CNTWD(4)), (WSTO, WSTO(3)) ,
 (DLYV,  DLXV(2)), (  DLZV, DLXV(3)) ,
 (RAN,RSUBA(2)),(RYN,RSUBA(3)),(RSUBU,RSUBA(4)),(UXN,USUBA(2)),
 (UYN,USUBA(3)),(USUBU,USUBA(4)),(RDDTV,RDDTU(2)),(RDDTW,
 RDDTU(3)),(RDTAN,RDOTA(2)),(RDTYN,RDOTA(3)),(RDOTV,
 RDOTA(4)),(UDTXN,UDOTA(2)),(UDTYN,UDOTA(3)),(UDOTU,UDOTA(4)),
 (RDOTU,RDOTA(4)),(IAXD,STARK),(IAXE,STARE),
 (DGR,CAPD(2)),(DDGR,CAPD(3)) $
```

```
EQUIVALENCE(LL,AA(2),DLTAX),(DLTAA(3),DLTAY),(DLTAA(4),DLTLO),
 ( TA (5), LL D), (DLTAA(6), DLT1 ) $
CO (N G, T, TF,  ,  , THGP, XLG, AXN,
HX, , , GC ,CGUNT,RA,P,RTP, ,S I,COSI,SINO,
CO  ,A ,CO ,C TS,ESP,E,A,M X,XNX,AXN ,
XN  ,UO,EO1,EO2,SI C,COSEO,ESI E,ECOSE, ,
SI ,COSU,  ,RTESG,LX, X,X,XDOT,RD T,RTA,RV CT,
X , , U , UZ TH, HO, XNUSQR, TS, C, DC,  V25, ZOVR ,
 RS R, RDOT, E X, LAT, XLONG, RKCNT,
OTPT,C TWD,STACD,OBSCD,SINPH,CAPD,CAPC,CAPS,
CAP ,O T,P IR ,XOVCT,SINTH,COSTH,THTA,TXYZ,
RHO ,R 1,S LH,TER S,WSTO,TIMTP,CSUBI,XLMPT ,XINCL,DLTAA,
RHO ,S  ,ABSDL,YLG ,
RC T,BSCNT, ESID,EFCNT,LSTAT,GATE2,EFLAG,
HOCNT, CNT,X1STSG, ES X,ONCT2,ETPFG,TIMJ,
CSU ,FU FT,ASUBX,DSUBX,XLSUFX,ESUPX,SSUBX,
ZSUBX,XLSUXH,ADOTU,ADDEL,ASUXT,DSUXT,FACTR,OTYPE,
XMNUTS,XLX,DELTX,A2OR,UMUO,RSUBA,ROVA,
USUBA, ENC , SS, OBRA1, OBRAN, TM , SATEL, STAID,
XLAMBA, ALPHA, DELTA, ITIMES, RANGE,
Q, RGC , XKAE, R2PR1, XK, SMIS,
SI T,T GRO,FLP25,VCO3,SIGS,SQRMU,XMU,
TH T,F,XKE,RHGO,CDC,XUMPRM,XKMPRM,
SIXP9, PR ,ONEPI,RADYN,XMPER,LINE,
DEN7, FLAG4, VFLAG, XMASS, TS, DC, VDATA, ISTORV,
RDOTU,RDOTA,UDOTA,RODTO,RHODT,EXOM,EYOM,RCUPE,
XMUA32,ABMX2,GATE1,GATE2,XJAY5,SUM2,RCNT2,
LSTCNT,SPHA,CPHA,SDEL,CCDEL,
ADEL,BDEL,CDEL,ABDEL,FFEL,HDELA,HDELB,HDELC,HDELX,CAPXX,
ARAY,XTVEC,YTVEC,ZTVEC,BXSS,BYSS,BZSS,BCOF,BAB,BIG,BBUF,CVERT,
AVEC, X EC, YVEC, ZVEC,ASIG1, IAXD, IBAUTH,
ISTORC,DLXV,RHOV,ADELV,C,DC,IMOVOB$
DIMENSION G(2),AXN(2),HX(3),FRAC(2),WX(3),AX(3),A(2),XMX(3),
XNX(3), XNI(2),XNODE(2),UO(2),UX(3),VX(3),X(3),XDOT(3),
 X X(3), E X(2) ,
CNT D(27),STACD(6),OBSCD(6),SINPH(2),CAPX(3),RHOX(3),TXYZ(4),
TER S(25), STO(25),TIMTP(6),CSUBI(6),X1STSG(9),ASUBX(3),
 DSUBX(3), XLSUBX(3), OTPT(8), DLTAA(6),
ESUBX(3),SSUBX(3),ZSUBX(3),XLSUXH(3),ADOTU(3),ASUXT(3),
DSUXT(3),XLX(3),DELTX(3),RSUBA(4),USUBA(4),
 RKSTO(79), HXI(3),
RKCNT(90), VDATA(1000), DLXV(3),
FUDFT(7),ARAY(6),XTVEC(6),YTVEC(6),ZTVEC(6),
RDOTU(3),RDOTA(4),UDOTA(4),RHODT(4),
BXSS(6),BYSS(6),BZSS(4),BCOF(20),BAB(3),
BIG(6),BBUF(26),CVERT(36),AVEC(3),XVEC(6),
YVEC(6),ZVEC(6), ASIG1(4),CAPXX(3), CAPD(3),
ADEL(3),BDEL(3),CDEL(3),ABDEL(3),FFEL(3),
HDELA(2),HDELB(2), HDELC(2),HDELX(3)$
P=(HXI**2+HYI**2+HZI** )$
ES =1. -(AXI**2+AYI**2+AZI**2)$
A=P/ES  RT =S RT(A(1))  $
XN=(SG  L*X E)/SQRT (A**3)$
RTP=SQ  ( )$
DO 1I=1,3$
WX(I)= I(I)/RT $
COS =W
SI I=S  T (1.0-COS **2) X Z=SI I$
XN X=(-  )  /X  XNY=X  X*WZ Y Y= X/X Z$
X X=(- Y)* Z$
AX  =  XI * X X +  YI * X Y $ GET COMP OF A ON N BAR
```

```
      AY  I=A   *X   +A   *X  Y+AZI*X ZS
      IF( X(  ))2,3,   $
      XNO E=  +AT   (  NY/  X)$
      GO TO 7
      IF( Y) ,ERROR,6$
      XNO E = 4.7123 8978 $        XNODE = (3/2) * PI
      GO TO 7
      XNODE=P OV2
      GO TO 7
      XNO E=ATAN (XNY/XNX)$
      XNODE(1)=PROPR(XNODE(1))$
7     U = RKSTO (5) - XNO E * SIGNF (1., 7) $   XL +- XNODE
      IF (SENSE LIGHT 1) 13,12 $
   12U0(1)=U$
   13  SENSE LIGHT 1 $
      U0(2) = U $      SOLVE KEPLERS EQUATION FOR E + OMEGA
      IF(U0(2)-TWOPI)15,15,14$
      U0(2)=U0(2)-TWOPI$
      GO TO 16$
      E01=U0(2)$
      DO 18I=1,30$
      COSEO=COS (E01)$
      SINEO=SIN (E01)$
      E02=AYNI*COSEO$
      ESINE=(SINEO*AXNI)-E02$
      E02=U0(2)+ESINE$
      IF (ABS (E02 - E01) - .000001) CNTRD2, CNTRD2, 17 $ TEST CNVRGNC
   17E01=E02     $
      CONTINUE$
      START TAC      $
      TMA      C/HLT,CWRD@ C/HLT,5$
      JMPL     6$
      ENDTAC     $
TRD2 ECOSE = SINEO * AYNI + COSEO * AXNI $
      R=(1.0-ECOSE)*A@AR=A/R@RTESQ=SQRT (ESQ)$
      XLGR=1.0+RTESQ COSU=ESINE/XLGR$
      SINU=((SINEO-AYNI)-(AXNI*COSU))*AR$
      COSU=(COSEO-AXNI+AYNI*COSU)*AR$
      DO 20 I=1,3$
      UX(I)=SINU*XMX(I)+COSU*XNX(I)$
      VX(I)=COSU*XMX(I)-SINU*XNX(I)$
20     X(I) = UX(I) * R$       COMPUTE POSITION VECTOR
      RVDOT=(RTA*SQR U)/R$
      RDOT=(RVDOT*ESINE)$
      RVDOT=RVDOT*RTESQ$
      DO 21 I=1,3$
21    XDOT(I)= VX(I)*RVDOT + UX(I) * RDOT$   COMPUTE VELOCITY VECTOR
       RETURN  $
      START TAC   $
RROR RUNOUT   $
      JMPL    3 $
I     F/3.141592653 $
 OPI F/6.283185306 $
IOV2 F/1.570796326 $
 RD  A/KEPLERS EQUATION NOT CONVERGING FOR E  $
      END TAC  $
      END$
      SUBROUTINE SETI(IAXD,I)$
      J=0                        $
    49J=J+1                      $
      IF(IAXD-J)55,51,50$
```

```
 50I=I+3                                 $
   GO TO 50                              $
 55START TAC$
   JMP      COEFF.TSTO55$
   REFOUT   COEFF.TSTOPS$
   PAGE   $
   END TAC$
 51RETURN                                $
   END                                   $
   SUBROUTINE HEAD (IPAGE)               $
   START TAC $
   SAVE     HDNG,0001PRO.HDNG $
   SAVE     CHED,0001PRO.CHED$
   END TAC $
   TABLEDEF HDNG(24),CHED(15) $
   WRITE OUTPUT TAPE 5, 6,(HDNG(I),I=1,10)$   WRITE 1ST LINE HEDING
6  FORMAT (2H70,3X,10A8)$
   WRITE OUTPUT TAPE 5,7,(HDNG(I),I=11,20),IPAGE,(CHED(I),I=1,15)$
7  FORMAT (2H10,10A8,6H  PAGE,I8/2H10/2H10,15A8) $
       RETURN$
   END       $
   SUBROUTINE PRINT(RKSTO,XLAT,XLONG,H) $
   AH = H / 1000. $
   WRITE OUTPUT TAPE 5,6,RKSTO,XLAT, XLONG, AH $
6  FORMAT (2H10,4(F15.7,5X))$
       RETURN$
   END       $
   SUBROUTINE PRINTO(STARC,TXYZ    ,RHO1,CAPA,SMLH)$
   WRITE OUTPUT TAPE 5,6,STARC,TXYZ,RHO1,CAPA,SMLH$
6  FORMAT(2H10,A8,4E15.8)$
       RETURN$
   END       $
    FUNCTION APROPR(X)$
* PROPERIZES ANGLE GIVEN IN DEGREES
   APROPR = 57.2957795 * PROPR(X/57.2957795) @ RETURN $
   END $
   FUNCTION PROPR(X)$
* PROPERIZES ANGLE GIVEN IN RADIANS
   START TAC$
   TMA      X$
   JAN      (P)+3H$
   FSM      TWOPI$
   JMP      (P)-1$
   FAM      TWOPI$
   JAN      (P)-1H$
   TAM      Y$
P TWOPI F/6.283185306                    $
   END TAC $
   PROPR = Y @ RETURN $
   END $
   EQUIVALENCE(RKSTO,RKCNT(2)),(HYI,RKCNT(11)),(HZ I,RKCNT(12)),
              (HXI,RKCNT(10)),(AXI,RKCNT(7)),(AYI,RKCNT(8)),
   (AZI,RKCNT(9)),(AYN,AX (2)),(HY,HX(2)),(HZ,HX(3)),
   (WY,WX( )),(WZ,WX(3)),(AY,AX(2)),(AZ,AX(3)),
   (X Y,X Y(2)),(XMZ,XMX(3)),(XNY,XNX(2)),(XNZ,XNX(3)),
   (AY I,  AI( )),(UY,UX(2)),(UZ,UX(3)),(VY,VX(2)),(VZ,VX(3)),
   (Y,X(2)),(Z,X(3)),(YDOT,XDOT(2)),(ZDOT,XDOT(3)),
   (X Y,X UX(2)),(XNUZ,X UX(3)),(YDTGR,XDTGR(2)),
   (Z TGR, DTG (3)),(YBDGR,XBDGR(2)),(ZBDGR,XBDGR(3)),
   (Y GR, GR( )),(ZDGR,X GR(3)),(AYGR,AXGR(2)),
   (AZGR, GR(3)),(EQY,EQX(2)),(EQZ,EQX(3)),(HYGR,HXGR(2)),
```

```
(HZCR, GR(2)),(COSTH,SINPH(2)),(CAPY,CAPX(2)),
(C Z, PX(3)),(RH ,R OX(2)),(RHOZ,RHOX(3)),
(X2 S ,X1STSG(2)),(X3 SG,X1STSG(3)),(X4THSG,X1STSG(4)),
(X5THSG,X1STSG(5)),(X6THSG,X1STSG(6)),(X7THSG,X1STSG(7)),
(X THSG,X1STSG(8)),(X9THSG,X1STSG(9)),(ASUBY,ASUBX(2)),
(ASUBZ, SUB (3)),(DSUBY,DSUBX(2)),(DSUBZ,DSUBX(3)),
(XLSUBY,XLSUBX(2)),(XLSUBZ,XLSUBX(3)),(ESUBY,ESUBX(2)),
(ESUBZ, SUB (3)),(SSUBY,SSUBX(2)),(SSUBZ,SSUBX(3)),
(ZSUBY,ZSUB (2)),(ZSUBZ,ZSUBX(3)),(XLSUYH,XLSUXH(2)),
(XLSUZH,XLSUXH(3)),(ADOTV,ADOTU(2)),(ADOTW,ADOTU(3)),
(ASUYT, SUXT(2)),(ASUZT,ASUXT(3)),(DSUYT,DSUXT(2)),
(DSUZT, SUXT(3)),(XLY,XLX(2)),(XLZ,XLX(3)),(DELTY,DELTX(2)),
(DELTZ, ELTX(3)),
(IRPT,C TWD(4)),(EPHEM,XXXX(25)),(IWSTO,WSTO(3)),
(DLYV, DLXV(2)), (  DLZV, DLXV(3)) ,
(RXN,RSUBA(2)),(RYN,RSUBA(3)),(RSUBU,RSUBA(4)),(UXN,USUBA(2)),
(UYN,USUBA(3)),(USUBU,USUBA(4)),(RDDTV,RDDTU(2)),(RDDTW,
RDDTU(3)),(RDTXN,RDOTA(2)),(RDTYN,RDOTA(3)),(RDOTV,
RDOTA(4)),(UDTXN,UDOTA(2)),(UDTYN,UDOTA(3)),(UDOTU,UDOTA(4)),
(RDOTU,RDOTA(4)),(IAXD,STARK),(IAXE,STARE),
(DGR,CAPD(2)),(DDGR,CAPD(3)) $
EQUIVALENCE(DLTAA(2),DLTAX),(DLTAA(3),DLTAY),(DLTAA(4),DLTLO),
 (DLTAA(5), DLTND), (DLTAA(6), DLTIN) $
COMMON TO, DT, TF, D, XM, THGR, XLO, AXN,
HX,XN,IPGCNT,COUNT,FRAC,P,RTP,WX,SINI,COSI,SINO,
COSO,AX,COND,CONTS,FSQ,E,A,XMX,XNX,AXNI,
XNODE,U,UO,E01,E02,SINEO,COSEO,ESINE,ECOSE,R,
SINU,COSU,AR,RTESQ,UX,VX,X,XDOT,RDOT,RTA,RVDOT,
XNUX, XNU , UZETH, RHO, XNUSQR, TS, C, DC, XOVR5, ZOVR5,
 RSQR, RRDOT, EQX, XLAT, XLONG, RKCNT,
OTPT,CNTWD,STACD,OBSCD,SINPH,CAPD,CAPC,CAPS,
CAPX,OALT,PHIRD,XOVCT,SINTH,COSTH,THTA,TXYZ,
RHOX,RHO1,SMLH,TERMS,WSTO,TIMTB,CSUBI,XLMPTH,XINCL,DLTAA,
RHOC,SUM,ARSDL,XLGR,
RCNT,BSCNT,RESID,EFCNT,LSTAD,GATE2,EFLAG,
BOCNT,O CNT,X1STSG,ARSHX,OMCT2,ETPEG,TIMJ,
CSUM,FO FT,ASUBX,DSUBX,XLSUBX,ESUBX,SSUBX,
ZSUBX,XLSUXH,ADOTU,ADDEL,ASUXT,DSUXT,FACTR,OTYPE,
XMNUTS,XLX,DELTX,A2OR,UMUO,RSUBA,ROVA,
USUBA, ENO , SS, OBRA1, OBRAN, TMM, SATEL, STAID,
XLAMBA, ALPHA, DELTA, ITIMES, RANGE,
Q, RGC , XMAE, R2PM1, XK, EMIS,
SIGHT,THGRO,FLP25,VCO3,SIGS,SQRMU,XMU,
THDOT,F,XKE,RHOO,CDC,XJMPRM,XKPRM,
SIXP9,H PRM,ONEPI,RADYN,XMPER,LINE,
DENT, F AG4, VFLAG, XMISS, TS, DC, VDATA, ISTORV,
RDDTU,RDOTA,UDOTA,RCDTO,RHODT,EXOM,EYOM,RCUPE,
XMUA32,ABMX2,GATE1,GATE2,XJAY5,SUM2,RCNT2,
LSTCNT,SPHA,CP A,SDFL,CCDEL,
ADEL,BDEL,CDEL,ABDEL,FLEL,HDELA,HDELB,HDELC,HDELX,CAPXX,
ARAY,XTVEC,YTVEC,ZTVEC,BXSS,BYSS,BZSS,BCOF,BAB,BIG,BBUF,CVERT,
AVEC, XVEC, YVEC, ZVEC,ASIG1, IAXD, IBOUTH,
ISTORC,DLYV,RHOV,ADELV,C,DC,IMOVOR$
DIMENSION W(2),AXN(2),HX(3),FRAC(2),WX(3),AX(3),A(2),XMX(3),
XNX(3), XMI(2),XNODE(2),UO(2),UX(3),VX(3),X(3),XDOT(3),
XNUX(3),XPTER(3),XBFGR(3),XFGR(3),AXGR(3),EQX(3 ),HXGR(3),
CNTWD(27),STACD(6),OBSCD(6),SINPH(2),CAPX(3),RHOX(3),TXYZ(4),
TERMS(25),A11(36),B11(12),OBUF(28),WSTO(25),TIMTB(6),CSUBI(6),
KNTFL(6),X1STSG(9),ASUBX(3),DSUBX(3),XLSUBX(3),OTPT(8),
         DLTAA(6),CXVERT(6),CYVERT(6),CZVERT(6),
ESUBX(3),SSUBX(3),ZSUBX(3),XLSUXH(3),ADOTU(3),ASUXT(3),
```

```
       DSL T(3),XL (3),DELTX(3),RSUBA(4),USUBA(4),
       XXX (1 0), PH  (10 4), RKSTO(79), HXI(3),
       HF   (2  ,  IG1(4)  ,
       RKL T(  ),C ED(15), A A(1000),RLXV(3),
       FU FT(7),A  Y( ),XTVEC(6),YTVEC(6),ZTVEC(6),
       RD TU( ),R  TA(4),U  T (4),RHOCT(4),
       BXSS(6),BYSS(6),BZSS(4),BCOF(20),BAB(3),
       BIG(6), BUF(26),CVECT(36),AVEC(3),XVEC(6),
       YVEC(6),ZVEC(6),BDIAG(25),BDIAS(25),
               QLM(25),
       SQBA(25),SQBB(25),SQBC(25),
       SQ  (625),S MB(625),SQ C(625),
       QLSQA(576),QLS P(24),QLSQX(24),
       ADE (3),BDEL(3),CDEL(3),ABDEL(3),FDEL(3),
       HDELA(2),HDELB(2), HDELC(2),HDELX(3),
       CAPXX(3),QLX(3),DELT(3,6),
       STARC(1000),CAPD(3) $
        DIMENSION IDBUF(10)$
       DIMENSION MCODE (3) $
       TABLE DEF TVOTC(12),CHED1(15),CHED2(15),CHED3(15),
       CHED4(15),OCHED(15),OBSHD(15),CHED5(15),THGROC(20)$
       TABLEDEF EQEQ(400) $
       READ ALL TAPES AHEAD
       WRITETAPE9,WSTO $
       WRITETAPE3,WSTO $
       WRITETAPE6,WSTO $
       WRITETAPE11,WSTO$
       START TAC  $
PI     F/3.141592653 $
PIOV2  F/1.570796326 $
CONV   F/.01745329251 $      DEGREES TO RADIANS
TWOPI  F/6.28318530 6 $
       TMR      N/10T23GH/8AT47$REWIND TAPE 10
       TIC       $
       JMP      (P)-1H$
       TMA      3 $
       TAM      SKIPSAV$
       TMA      C/TUM,SUBER16 C/JMP, SUBER $
       TAM      3$
       JMP      STARTO  $
AFRNTH W/           $
 NPCA  F/144  . $
 GTR   F/.25 $
  PTI  F/.43752691 -2$
 FFSQ  F/6.766580E-3      $
THGROC F/0.0            $ 1958
       F/0.0            $ 1959
       F/98.67 01       $ 1960
       F/99.42 937      $ 1961
       F/99.1822167     $ 1962
       F/98.943500314 $ 1963
       F/98.7 47917 $ 1964
       F/0.0            $ 1965
       F/0.0            $ 1966
       F/0.0            $ 1967
       F/0.0            $ 1968
       F/0.0            $ 1969
       F/0.0            $ 1970
```

```
          F/0.0                 $ 1971
          F/0.0                 $ 1972
          F/0.0                 $ 1973
          F/0.0                 $ 1974
          F/0.0                 $ 1975
          F/0.0                 $ 1976
          F/0.0                 $ 1977
QEQ       F/-0.9[illegible]1         $ NOV 1 1962
          F/-0.97[illegible]         $
          F/-0.97[illegible]         $
          F/-0.97[illegible]         $
          F/-0.96[illegible]         $
          F/-0.967             $
          F/-0.969             $
          F/-0.973             $
          F/-0.979             $
          F/-0.98[illegible]         $
          F/-0.98[illegible]         $
          F/-0.9[illegible]          $
          F/-0.98[illegible]         $
          F/-0.981             $
          F/-0.97[illegible]         $
          F/-0.97[illegible]         $
          F/-0.96[illegible]         $
          F/-0.96[illegible]         $
          F/-0.961             $
          F/-0.961             $
          F/-0.962             $
          F/-0.966             $
          F/-0.970             $
          F/-0.97[illegible]         $
          F/-0.98[illegible]         $
          F/-0.981             $
          F/-0.98[illegible]         $ NOV27 1962
          F/-0.975             $ NOV28 1962
          F/-0.97[illegible]         $
          F/-0.96[illegible]         $
          F/-0.9[illegible]          $
          F/-0.95[illegible]         $
          F/-0.94[illegible]         $
          F/-0.94[illegible]         $
          F/-0.949             $
          F/-0.952             $
          F/-0.953             $
          F/-0.95[illegible]         $
          F/-0.957             $
          F/-0.95[illegible]         $
          F/-0.94[illegible]         $
          F/-0.940             $
          F/-0.931             $
          F/-0.92[illegible]         $
          F/-0.91[illegible]         $
          F/-0.91[illegible]         $
          F/-0.91[illegible]         $
          F/-0.91[illegible]         $
          F/-0.91[illegible]         $
          F/-0.921             $
          F/-0.923             $
          F/-0.925             $
          F/-0.925             $
          F/-0.923             $ DEC24 1962
```

```
F/-0.91?    $ DEC25 1962
F/-0.913    $
F/-0.9      $
F/- .89     $
F/-0.89     $
F/- .8      $
F/- .8      $  DEC31 1962
F/-0.8      $  JAN 1 1963
F/-0.8      $
F/-0.89     $
F/-0.89     $
F/-0.89     $
F/-0.89     $
F/-0.88     $
F/-0.879    $
F/-0.871    $
F/-0.86     $
F/-0.853    $
F/-0.85     $
F/-0.853    $ JAN13 1963
F/-0.855    $ JAN14 1963
F/-0.858    $
F/-0.861    $
F/-0.865    $
F/-0.868    $
F/-0.873    $
F/-0.87     $
F/-0.868    $
F/-0.86     $
F/-0.858    $
F/-0.853    $
F/-0.847    $
F/-0.843    $
F/-0.842    $
F/-0.843    $
F/-0.847    $
F/-0.85     $
F/-0.857    $
F/-0.86     $ FEB 1 1963
F/-0.861    $
F/-0.859    $
F/-0.855    $
F/-0.850    $
F/-0.845    $
F/-0.841    $ FEB 7 1963
F/-0.839    $ FEB 8 1963
F/-0.84     $
F/-0.843    $
F/-0.848    $
F/-0.85     $
F/-0.861    $
F/-0.867    $
F/-0.872    $
F/-0.875    $
F/-0.877    $
F/-0.87     $
F/-0.87     $
F/-0.871    $
F/-0.867    $
F/-0.865    $
F/-0.86     $
```

```
F/-0.86[illegible]   $
F/-0.873   $
F/-0.881   $
F/-0.8[illegible]   $
F/-0.89[illegible]   $
F/-0.899   $
F/-0.899   $
F/-0.899   $
F/-0.895   $ MAR 4 1963
F/-0.892   $ MAR 5 1963
F/-0.890   $
F/-0.890   $
F/-0.892   $
F/-0.897   $
F/-0.903   $
F/-0.911   $
F/-0.919   $
F/-0.928   $
F/-0.935   $
F/-0.940   $
F/-0.944   $
F/-0.946   $
F/-0.946   $
F/-0.944   $
F/-0.942   $
F/-0.941   $
F/-0.941   $
F/-0.944   $
F/-0.949   $
F/-0.957   $
F/-0.965   $
F/-0.973   $
F/-0.978   $
F/-0.980   $ MAR29 1963
F/-0.980   $ MAR30 1963
F/-0.977   $
F/-0.974   $
F/-0.971   $
F/-0.971   $
F/-0.973   $
F/-0.976   $
F/-0.982   $
F/-0.990   $
F/-0.998   $
F/-1.006   $
F/-1.013   $
F/-1.018   $
F/-1.022   $
F/-1.023   $
F/-1.023   $
F/-1.021   $
F/-1.019   $
F/-1.016   $
F/-1.015   $
F/-1.016   $
F/-1.019   $
F/-1.024   $
F/-1.031   $
F/-1.038   $ APR23 1963
F/-1.043   $ APR24 1963
F/-1.045   $
```

```
F/-1.043     $
F/-1.037     $
F/-1.043     $
F/-1.027     $
F/-1.025     $
F/-1.024     $
F/-1.025     $
F/-1.025     $
F/-1.030     $
F/-1.043     $
F/-1.046     $
F/-1.051     $
F/-1.055     $
F/-1.056     $
F/-1.056     $
F/-1.054     $
F/-1.050     $
F/-1.045     $
F/-1.041     $
F/-1.039     $
F/-1.036     $
F/-1.036     $
F/-1.039     $ MAY18 1963
F/-1.043     $ MAY19 1963
F/-1.047     $
F/-1.051     $
F/-1.052     $
F/-1.053     $
F/-1.044     $
F/-1.037     $
F/-1.029     $
F/-1.022     $
F/-1.017     $
F/-1.015     $
F/-1.016     $
F/-1.019     $
F/-1.022     $
F/-1.026     $
F/-1.030     $
F/-1.032     $
F/-1.032     $
F/-1.031     $
F/-1.027     $
F/-1.022     $
F/-1.014     $
F/-1.009     $
F/-1.004     $
F/-1.000     $ JUN12 1963
F/-0.993     $ JUN13 1963
F/-0.991     $
F/-1.001     $
F/-1.004     $
F/-1.007     $
F/-1.009     $
F/-1.007     $
F/-1.001     $
F/-0.993     $
F/-0.984     $
F/-0.974     $
F/-0.969     $
F/-0.965     $
```

F/-0.964 $
F/-0.965 $
F/-0.968 $
F/-0.971 $
F/-0.975 $
F/-0.977 $
F/-0.979 $
F/-0.977 $
F/-0.974 $
F/-0.969 $
F/-0.962 $
F/-0.956 $ JUL 7 1963
F/-0.950 $ JUL 8 1963
F/-0.946 $
F/-0.943 $
F/-0.944 $
F/-0.946 $
F/-0.950 $
F/-0.954 $
F/-0.957 $
F/-0.956 $
F/-0.953 $
F/-0.947 $
F/-0.94 $
F/-0.932 $
F/-0.926 $
F/-0.922 $
F/-0.921 $
F/-0.922 $
F/-0.926 $
F/-0.931 $
F/-0.936 $
F/-0.940 $
F/-0.943 $
F/-0.944 $
F/-0.943 $
F/-0.940 $ AUG 1 1963
F/-0.936 $ AUG 2 1963
F/-0.931 $
F/-0.927 $
F/-0.923 $
F/-0.922 $
F/-0.923 $
F/-0.927 $
F/-0.933 $
F/-0.939 $
F/-0.940 $
F/-0.947 $
F/-0.945 $
F/-0.943 $
F/-0.939 $
F/-0.933 $
F/-0.929 $
F/-0.925 $
F/-0.927 $
F/-0.930 $
F/-0.937 $
F/-0.94 $
F/-0.949 $
F/-0.956 $
F/-0.961 $

```
F/-0.965     $ AUG26 1963
F/-0.967     $ AUG27 1963
F/-0.967     $
F/-0.966     $
F/-0.963     $
F/-0.961     $
F/-0.959     $
F/-0.959     $
F/-0.961     $
F/-0.966     $
F/-0.97[illegible]     $
F/-0.9[illegible]     $
F/-0.989     $
F/-0.99[illegible]     $
F/-0.996     $
F/-0.995     $
F/-0.993     $
F/-0.989     $
F/-0.986     $
F/-0.985     $
F/-0.985     $
F/-0.990     $
F/-0.996     $ SEP17 1963
F/-1.004     $ SEP18 1963
F/-1.013     $
F/-1.021     $
F/-1.029     $
F/-1.034     $
F/-1.037     $
F/-1.039     $
F/-1.039     $
F/-1.038     $
F/-1.036     $
F/-1.035     $
F/-1.035     $
F/-1.037     $
F/-1.042     $
F/-1.049     $
F/-1.057     $
F/-1.065     $
F/-1.071     $
F/-1.07[illegible]     $
F/-1.074     $
F/-1.071     $
F/-1.067     $
F/-1.063     $
F/-1.061     $
F/-1.061     $
F/-1.06[illegible]     $
F/-1.069     $ OCT14 1963
F/-1.07[illegible]     $ OCT15 1963
F/-1.0[illegible]     $
F/-1.0[illegible]     $
F/-1.09[illegible]     $
F/-1.1[illegible]     $
F/-1.1[illegible]     $
F/-1.1[illegible]     $
F/-1.1[illegible]7     $
F/-1.1[illegible]     $
F/-1.1[illegible]     $
F/-1.1[illegible]     $
```

```
F/-1.09[illegible]   $
F/-1.05              $
F/-1.1[illegible]    $
F/-1.10              $
F/-1.111             $
F/-1.11[illegible]   $
F/-1.123             $
F/-1.125             $
F/-1.12[illegible]   $
F/-1.11              $
F/-1.113             $
F/-1.106             $
F/-1.100             $
F/-1.097             $
F/-1.097             $
F/-1.099             $ NOV10 1963
F/-1.103             $ NOV11 1963
F/-1.10[illegible]   $
F/-1.11[illegible]   $
F/-1.119             $
F/-1.122             $
F/-1.123             $
F/-1.122             $
F/-1.12[illegible]   $
F/-1.115             $
F/-1.11[illegible]   $
F/-1.105             $
F/-1.101             $
F/-1.09[illegible]   $
F/-1.096             $
F/-1.09[illegible]   $
F/-1.101             $
F/-1.105             $
F/-1.109             $
F/-1.11[illegible]   $
F/-1.109             $
F/-1.10[illegible]   $
F/-1.093             $
F/-1.083             $
F/-1.07[illegible]   $
F/-1.067             $
F/-1.063             $
F/-1.062             $ DEC 7 1963
F/-1.06[illegible]   $ DEC 8 1963
F/-1.067             $
F/-1.07[illegible]   $
F/-1.073             $
F/-1.07[illegible]   $
F/-1.07[illegible]   $
F/-1.07[illegible]   $
F/-1.06[illegible]   $
F/-1.06[illegible]   $
F/-1.05[illegible]   $
F/-1.04[illegible]   $
F/-1.04[illegible]   $
F/-1.037             $
F/-1.03[illegible]   $
F/-1.03              $
F/-1.03[illegible]   $
F/-1.0[illegible]    $
F/-1.04[illegible]   $
```

```
F/-1.042     $
F/-1.041     $
F/-1.036     $
F/-1.023     $
F/-1.01[illegible]     $
F/-1.017     $ DEC31 1963
F/-0.999     $ JAN 1 1964
F/-0.997     $
F/-0.99[illegible]     $
F/-0.992     $
F/-0.993     $
F/-0.99[illegible]     $
F/-0.999     $
F/-1.001     $
F/-1.002     $
F/-1.001     $
F/-0.997     $
F/-0.993     $
F/-0.987     $
F/-0.981     $
F/-0.975     $
F/-0.970     $
F/-0.969     $
F/-0.967     $
F/-0.970     $
F/-0.973     $
F/-0.970     $
F/-0.981     $
F/-0.983     $
F/-0.981     $
F/-0.976     $
F/-0.962     $
F/-0.960     $
F/-0.957     $
F/-0.942     $
F/-0.946     $
F/-0.947     $
F/-0.951     $ FEB 1 1964
F/-0.956     $
F/-0.961     $
F/-0.966     $
F/-0.970     $
F/-0.971     $
F/-0.971     $
F/-0.969     $
F/-0.966     $
F/-0.962     $
F/-0.958     $
F/-0.956     $
F/-0.955     $
F/-0.957     $
F/-0.961     $
F/-0.967     $
F/-0.976     $
F/-0.980     $
F/-0.985     $
F/-0.986     $
F/-0.985     $
F/-0.981     $
F/-0.976     $
F/-0.971     $
```

```
F/-0.96        $
F/-0.96[illegible]       $
F/-0.97        $
F/-0.97[illegible]       $
F/-0.9[illegible]7       $
F/-0.99        $ MAR 1 1964
F/-0.99        $
F/-1.004       $
F/-1.00[illegible]       $
F/-1.011       $
F/-1.012       $
F/-1.011       $
F/-1.010       $
F/-1.00[illegible]       $
F/-1.007       $
F/-1.007       $
F/-1.010       $
F/-1.015       $
F/-1.022       $
F/-1.03[illegible]       $
F/-1.03[illegible]       $
F/-1.045       $
F/-1.04[illegible]       $
F/-1.049       $
F/-1.047       $
F/-1.043       $
F/-1.032       $
F/-1.036       $
F/-1.036       $
F/-1.039       $
F/-1.044       $
F/-1.052       $
F/-1.060       $
F/-1.069       $
F/-1.075       $
F/-1.081       $
F/-1.084       $ APRIL 1 1964
F/-1.085       $
F/-1.085       $
F/-1.084       $
F/-1.082       $
F/-1.081       $
F/-1.080       $
F/-1.081       $
F/-1.085       $
F/-1.091       $
F/-1.092       $
F/-1.106       $
F/-1.113       $
F/-1.116       $
F/-1.117       $
F/-1.113       $
F/-1.109       $
F/-1.102       $
F/-1.098       $
F/-1.095       $
F/-1.096       $
F/-1.100       $
F/-1.105       $
F/-1.112       $
F/-1.119       $
```

```
F/-1.12[illegible]     $
F/-1.12[illegible]     $
F/-1.13[illegible]     $
F/-1.13[illegible]     $
F/-1.13[illegible]     $
F/-1.127     $ [illegible]Y 1 19[illegible]4
F/-1.123     $
F/-1.12      $
F/-1.117     $
F/-1.11[illegible]     $
F/-1.11[illegible]     $
F/-1.119     $
F/-1.12[illegible]     $
F/-1.13[illegible]     $
F/-1.135     $
F/-1.13[illegible]     $
F/-1.137     $
F/-1.133     $
F/-1.125     $
F/-1.117     $
F/-1.[illegible]09     $
F/-1.[illegible]03     $
F/-1.[illegible]06     $
F/-1.[illegible]     $
F/-1.[illegible]     $
F/-1.[illegible]     $
F/-1.[illegible]13     $
F/-1.[illegible]17     $
F/-1.[illegible]19     $
F/-1.[illegible]2[illegible]     $
F/-1.[illegible]1[illegible]     $
F/-1.[illegible]15     $
F/-1.[illegible]1[illegible]     $
F/-1.[illegible]     $
F/-1.099     $
F/-1.09[illegible]     $
F/=1.[illegible]9[illegible]     $ [illegible]UNE 1 1964
F/-1.[illegible]88     $
F/-1.[illegible]8[illegible]     $
F/-1.[illegible]91     $
F/-1.[illegible]9[illegible]     $
F/-1.[illegible]9[illegible]     $
F/-1.101     $
F/-1.10[illegible]     $
F/-1.09[illegible]     $
F/-1.08[illegible]     $
F/-1.077     $
F/-1.067     $
F/-1.05[illegible]     $
F/-1.05[illegible]     $
F/-1.05      $
F/-1.051     $
F/-1.05[illegible]     $
F/-1.05[illegible]     $
F/-1.06[illegible]     $
F/-1.06      $
F/-1.06[illegible]     $
F/-1.06      $
F/-1.05[illegible]     $
F/-1.05[illegible]     $
F/-1.0[illegible]      $
```

```
F/-1.0[illegible]      $
F/-1.033      $
F/-1.025      $
F/-1.026      $
F/-1.025      $
F/-1.027      $ JULY 1 1964
F/-1.030      $
F/-1.033      $
F/-1.035      $
F/-1.037      $
F/-1.034      $
F/-1.028      $
F/-1.010      $
F/-1.009      $
F/-0.999      $
F/-0.992      $
F/-0.989      $
F/-0.989      $
F/-0.992      $
F/-0.996      $
F/-1.000      $
F/-1.003      $
F/-1.005      $
F/-1.004      $
F/-1.002      $
F/-0.99[illegible]      $
F/-0.993      $
F/-0.982      $
F/-0.983      $
F/-0.979      $
F/-0.977      $
F/-0.97[illegible]      $
F/-0.981      $
F/-0.985      $
F/-0.990      $
F/-0.995      $
F/-0.99[illegible]      $  AUG 1 1964
F/-0.999      $
F/-0.99[illegible]      $
F/-0.99[illegible]      $
F/-0.983      $
F/-0.974      $
F/-0.96[illegible]      $
F/-0.965      $
F/-0.963      $
F/-0.97[illegible]      $
F/-0.975      $
F/-0.98[illegible]      $
F/-0.98[illegible]      $
F/-0.99[illegible]      $
F/-0.99[illegible]      $
F/-0.995      $
F/-0.993      $
F/-0.991      $
F/-0.987      $
F/-0.98[illegible]      $
F/-0.98[illegible]      $
F/-0.9[illegible]      $
F/-0.9[illegible]      $
F/-0.980      $
F/-0.995      $
```

```
F/-1.00[illegible]     $
F/-1.0[illegible]      $
F/-1.01[illegible]     $
F/-1.01       $
F/-1.01[illegible]     $
F/-1.0[illegible]      $
F/-1.01[illegible]     $ SEPT 1 1964
F/-1.0        $
F/-1.0[illegible]1     $
F/-0.94[illegible]     $
F/-1.0[illegible]      $
F/-1.0[illegible]      $
F/-1.013      $
F/-1.021      $
F/-1.02[illegible]     $
F/-1.035      $
F/-1.04[illegible]     $
F/-1.043      $
F/-1.043      $
F/-1.043      $
F/-1.041      $
F/-1.039      $
F/-1.03[illegible]     $
F/-1.03[illegible]     $
F/-1.041      $
F/-1.046      $
F/-1.053      $
F/-1.061      $
F/-1.07[illegible]     $
F/-1.077      $
F/-1.08[illegible]     $
F/-1.0[illegible]7     $
F/-1.081      $
F/-1.07[illegible]     $
F/-1.073      $
F/-1.0[illegible]      $
F/-1.06[illegible]     $ OCT 1 1964
F/-1.06       $
F/-1.07[illegible]     $
F/-1.079      $
F/-1.087      $
F/-1.09[illegible]     $
F/-1.0[illegible]3     $
F/-1.0[illegible]      $
F/-1.011      $
F/-1.012      $
F/-1.011      $
F/-1.009      $
F/-1.007      $
F/-1.0[illegible]      $
F/-1.0[illegible]      $
F/-1.0[illegible]      $
F/-1.0[illegible]      $
F/-1.01[illegible]     $
F/-1.121      $
F/-1.12[illegible]     $
F/-1.1[illegible]      $
F/-1.1[illegible]      $
F/-1.14[illegible]     $
F/-1.13[illegible]     $
F/-1.133      $
```

F/-1.12[illegible]	$	
F/-1.12	$	
F/-1.11[illegible]	$	
F/-1.110	$	
F/-1.11	$	
F/-1.12[illegible]	$	
F/-1.12[illegible]	$	NOV 1 1963
F/-1.1[illegible]	$	
F/-1.13[illegible]	$	
F/-1.142	$	
F/-1.144	$	
F/-1.143	$	
F/-1.141	$	
F/-1.137	$	
F/-1.132	$	
F/-1.129	$	
F/-1.125	$	
F/-1.123	$	
F/-1.123	$	
F/-1.125	$	
F/-1.129	$	
F/-1.133	$	
F/-1.13[illegible]	$	
F/-1.141	$	
F/-1.141	$	
F/-1.137	$	
F/-1.130	$	
F/-1.120	$	
F/-1.110	$	
F/-1.102	$	
F/-1.097	$	
F/-1.095	$	
F/-1.097	$	
F/-1.100	$	
F/-1.10[illegible]	$	
F/-1.107	$	
F/-1.109	$	DEC 1 1964
F/-1.10[illegible]	$	
F/-1.107	$	
F/-1.103	$	
F/-1.097	$	
F/-1.09[illegible]	$	
F/-1.083	$	
F/-1.077	$	
F/-1.073	$	
F/-1.07[illegible]	$	
F/-1.07[illegible]	$	
F/-1.071	$	
F/-1.07[illegible]	$	
F/-1.077	$	
F/-1.079	$	
F/-1.07[illegible]	$	
F/-1.07[illegible]	$	
F/-1.06[illegible]	$	
F/-1.057	$	
F/-1.04[illegible]	$	
F/-1.03[illegible]	$	
F/-1.0[illegible]	$	
F/-1.02	$	
F/-1.021	$	
F/-1.022	$	

```
          F/-1.02      $
          F/-1.02      $
          F/-1.02      $
          F/-1.02      $
          F/-1.02      $
          F/-1.02      $   DEC 31 1964
          END TAC      $
*
*
 START  NSTART=      $
 START  NSTART= START-1$
        IF(NSTART)FINIS,START2,START2              $
 START2 IXXX=1  4 $
        IHED=0 $
        XFTPM=3.2808333333       $ FEET PER METER              328083333
        XFTNM=6080.2             $ FEET PER NAUTICAL MILE      6080.2
        XMPNM=XFTNM/XFTPM        $ METERS PER NAUT MILE
        XMPER=6378165.0          $ METERS PER EARTH RADIUS  6378165
        XAXS=XMPER/XMPNM         $ NAUT MILES PER EARTH RADIUS
        PASTEKA = 0.0  $
        NTAPE6=  $
        NTAPE3=  $
*
*       GO TO FRANK BROWNS DATA READ$
        DO(EOFR) I=1,7 $
 EOFR   CNTWD(I)=0.0$ SET CNTWD ARRAY TO ZERO
        CALL STARTER(HCNG,CNTWD,X1STSG,TO,DT,TF,D,XM,XLO,AXN,HX,IPNTFL,
        ABSMX,  CODE,ORGDA,ORGTM,REFDA,REFTM,ABMX2,EFLAG,IXYR,XAXS)$
         RADYN = 57.2957795$          SET UP CONSTANTS
         XK = .2504742E10 $
        R2P 1=.3989422E1$
        PASIFKA = 0.0 $
        IF(IXYR-5)IS1,IS2,IS1    $
 IS1    IF(IXYR-6)IS3,IS4,IS3    $
 IS3    IF(IXYR-7)IS5,IS8,IS5  $
 IS5    WRITE OUTPUT TAPE 5,IS6,IXYR $
 IS6    FORMAT(2H10,5HYEAR ,I2,36H GIVEN AS INPUT NOT IN EQUINOX TABLE)$
        START T C   $
        RUNOUT      $
        JUMP     5  $
        END TAC     $
 IS2    IT=ORGTM-304   $
        GO TO IS7  $
 IS4    IT=ORGTM+61$
        GO TO IS7  $
 IS8    IT=ORGTM+426 $
 IS7    CONTINUE    $
        RKOUT=1.0   $
        EMISS=.
        STANT=. 8560728
        THETC=ORGROT(IXYR)
        FLK 5=-.2507844A$
        VO  =.40401762596E16$
        SIGS=.  72  -4$
        SQRT E=1.0$
        XM =1.0
        THETAT=.0588344  $
        F=. 03452329 9$
        RH =1.3259
        XKE=.074365129              $
        CDO=.9   $
```

```
        X.MPRM=1.62341E-3$
        H PRM=-6.E-6$
        XKMPRM=5.39E-6$
        XJAY5=-0.2E-6
        SIXP9=6.972E
        ONEPI=1.17391
        CD=2.0   $
        AE=1.0
START2  APR    =  0.   $
        APRN 56 = 0.0 $
        APRNT71 = 0.0 $
        APRN 86 = 0.0 $
        APRN 03 = 0.0 $
        APRN 13=0.0$
        APRNTS=APRNTS $
        NTAPE2X= 100 $
        IPAG  =    $
        DRGCO=XMFER*(D**2/XM)*(CD*RHOO/2.0)*PI**2* (-1.0)  $
        FRAC(1)=DRGTM/XMNPDA$FRAC(2)=SIDRT*FRAC(1)$
        FRAC(2)=360. *FRAC(1)+FRAC(2)$
        THGR = APROPR (ORGDA * SIDRT + FRAC(2) + THGRO ) $
         THGRD = CONV * THGR $
        START TAC $
        TMA      CNTWD
        JAZ      (P)+2H$
        JMP      SEC2       $
        TMD      /0      $
        TOM      CNTWD   .
        RND TAC
SE      CNTWD(10)=(REFDA-ORGDA)*XMNPDA+REFTM-ORGTM$  NEW-OLD EPOCH, INS
        IF(CNTWD(10)*DT) LT(0.),GO TO SEC3D$  =0 IF CANNOT BE REACHED
        IF(  (TF)) GT(ABS(CNTWD(10))),GO TO SEC3$    IN SPECIFIED
 SEC3D  RIT  OUTPUT TAPE 5,SEC2D$                         INTEGRATIO
SEC2D   FORMAT(18H2UNO   START TIME //)$
         CNTWD(10) = 0.0    $
SE      REWIND6 $
        REWIND 11    $
        TPF6 =0.0 $
        NTAPE6=0 $
        GATE1=ABSMX$
        GATE2=ABMX2$
        OMCNT=1.0MO CT2=0.0$
        THE FOLLOWING TRANSFER OF COLUMN HEADINGS MAY BE CHANGED$
         O 1. I=1,15$
         HED(I) = CHED1(I) $  TIME-LATITUDE-LONGITUDE - ALTITUDE TO HEAD
   COMPUTE FOR EPOCH, VALUES OF -
    INCLINATION, NODE,   BAR, MEAN MOTION, A BAR
BEGIN    IF( 9THSG) GTE(0.), X9THSG = 3.$
          R  I    3$
        CNTWD(12)=0.0$CNTWD(13)=0.0$
        ITIMES=X9THSG$
        P=HX**2+HY**2+HZ**2$
        RTP=SQRT(P)$
        WX=HX/RTP$WY=HY/RTP$WZ=HZ/RTP$
        SINI=SQRT(1.0-WZ**2)$
        SINO=WX/SINI$COSO=(-WY/SINI)$
        COSI=WZ$
        IF(COSI) E ( .), XINCL = PIOV2$ GO TO 19$
          XINCL = ATAN( SINI/ COSI)  $
         IF(XINCL) LT (0.), XINCL = PI + XINCL$
   19   IF (COSO) E (0.), XNODE(2) = SIGNF(PIOV2,SINI)$ GO TO 23 $
```

```
        X CDE( ) = ATA (SI C / COSC)    $
        IF (COS ) LT ( .), X OGE (2) = XNOCF(2) + PI $
3       X  FE (2) = F (PR (XNOFE(2)) +
        AX=(AX COS -C SI*AY *SINO)$
        AY=(AX *SI +C SI*AY *COSO)$
        AZ=(AY *SI I)$
        CO =(- K*D**2/XN)  $
        COUTS = (VC 3*  .25)/(E IS*STGS) $
        ES =1. -(AX**2+AY**2+AZ**2)$
        A(1)= F/ESQ & A(2)= A(1)    $
        XN= XKE /( SQRT (A(1))*A)*SQRMU  $
        NCNTRL=1$
        LI = 55$
        COUNT= .0$
        IP T=  $
        AX I=AX (1) AY I=AY $
*  SET UP TO INTEGRATE EPHEMERIS
        RKSTO(1)=TO$
        RKSTO(4)=DT RKSTO(5)=XLO$
        DO 26 I=1,3$
        RKSTO(I+5)=AX(I)$
 26     RKSTO(I+8) = X(I)$
        CALL XYZSR$
        XNOZ=Z  T             $
        A2OR=AR*ARRSUB =A2OR*ESINE$
        UMU =(U-UO)*(-3.0/2.0)$
        RSUBA=( MUO*RS BU)+2$
        ROVA=R/ $
        RX =(A  I-COSE )*A2OR$
        RY =(AY I-SINE )*A2OR$
        US U=  ESQ*A2OR$
        US A=USURU*UM O$
        DE  =X GR**2* TESQ$
        TE =(X GR*ECOSF)*(-1.0)-ESQ+(1.0)$
        UY =(TE P/DENO )*ESINE$
        UX =UY  +AX I$
        RO A=1. +ROVA$
        UX =UX +ROV *SINEO$
        UX =((- YNI/XLGR)+UXN)*A2OR$
        UY =UY +AY I-R VA*COSE $
        UY=A2O *((AXNI/XLGR)+UYN)$
*     WRITE PARA ETERS FOR STD DEVIATION DETERMINATION
        WRITE T PE 11, KSTO(1),(RSUPA(I),I=1,4),(USUBA(I),I=1,4),
        (UX(I),I=1,3),(VX(I),I=1,3),R,COSI,SINI,SINU,COSU,
        (X(I),I=1,3),( X(I),I=1,3)   $
         IF(EFL G)27,S C3F,27   $
     27 NC TRL= XXX $
        SENSE LIGHT 4$
        GO TO C TRL$
 SEC3F  RECTI = 0.0 $
        NTAB E3=  $
        START T C$
        TMA      C/U P, FRIV C/UVP,C TRL $
        T        F/R CN ,T15  /7T39 C/0T47     $
        UMP      RUN3F , RUN F$            GO INTEGRATE EPHEMERIS
        UMP      RKF            $
        UMP      CSE D                    $
        PA       $
*       ERI  C  PUT S   LUES OF DERIVATIVES WITH RESPECT TO TIME OF
*       A  R  A,    A  R,         ------       GIVEN A BAR, H BAR, AND XL
    IV  TJ       UT RV
```

```
        EN  TAG
        CALL  XI  SF
        XM   =Z   T$
*   CO  UT  PE TURB TI E ACCELERATIONS
*   I.E.,       LEAVE OUT DRAG  UNTIL RECTIFY ROUTINE IS CALLED, AND
*         C  PUT  BULGE PERTURBATION
 B LGE  RS  =R*  R5=RS R*RS R*  $
        UZS R=  *UZ  XOVR5=X/R5 ZOVR5=Z/R5$
        XB  R=  VR5*(5.0*UZSQR-1.0)*XJMPRM$
        XB  R=X DGR+(X VR5*Z*H PRM/RSQR)*(7.0*UZSQR-3.0)$
        XB  R=X DGR+((XOVR5*XK PRM)/(6.0*RSQR))*(42.0*UZSQR
        -63.0*U SQR*UZSQR-3.0)$
        XBDGR=X DGR+((21.0*XJAY5*XOVR5*UZ)/(8.0*RSQR*R))*(5.0
        -30.0*UZSQR+33.0*UZSQR*UZSQR)$
        YBDGR=X DGR*(Y/X)$
        ZBDGR=Z VR5*(5.0*UZSQR-3.0)*XJMPRM$
        ZBDGR=Z DGR+(.6*HMPRM/R5)*(35.0*UZSQR*UZSQR/3.0
        -10.0*UZSQR+1.0)$
        ZBDGR=Z DGR+((ZOVR5*XK PRM)/(6.0*RSQR))*(-63.0*UZSQR
        *UZSQR+70.0*UZSQR-15.0)$
        ZBDGR=Z DGR+((3.0*XJAY5)/(8.0*R5*RSQR))*(-5.0
        +UZSQR*(105.0+UZSQR*(-315.0+UZSQR*231.0)))$
        THGRM=CONV*(THGPO+SIDRT*ORGDA+(TWOPI+SIDRT)*RKSTO(4)/XMNPDA)$
        UXTIME=ORGDA+RKSTO(4)/XMNPDA$
        CALL BELGE(IXYR,UXTIME,X,THGRM,AAA,BBB,CCC)$
        XBDGR=X DGR+AAA$
        YBDGR=Y DGR+BBB$
        ZBDGR=Z DGR+CCC$
        CALL PRESS(IXYR,X,UXTIME,D,XM,VV,U1,U2,U3)$
        XBDGR=X DGR+VV*U1$
        YBDGR=Y DGR+VV*U2$
        ZBDGR=Z DGR+VV*U3$
        DETERMI E TOTAL PERTURBATIONS
        XDGR= X DGR(1) $
        YDGR=YB GR$
        ZDGR=ZB GR$
        RRDOT=X*XDOT+Y*YDOT+Z*ZDOT$
        RRDGR=X*XDGR+Y*YDGR+Z*ZDGR$
        RDRDG=X GR*XDOT+YDGR*Y OT+ZDGR*ZDOT$
        CAP =RR OT/SQR U  $
        DGR=     RRDGR/SQRMU$
        DDGR=RDRDG/SQRMU$
        DDGR=DDGR*2.0 $
*   RELATE PERTURBATIO S TO OR ITAL PARAMETERS
        XNGR = (DDGR/SQPMU)*A*XN*(-1.5)@RBDGR=0.0$
        DO 29 I=1,3$
 29     RBDGR = XDGR(I) * WX(I) + RBDGR$
        SMLGR=SQRMU*(WZ+1.0)*RTP$
        SMLGR=(RBDGR/S LGR)*7$
 E XYZ  EQX=WY*AZI- Z*AYI$
        EQY=WZ*AXI- X* ZI$
        EQZ=WX* YI- Y*AXI$
        E2VGR=0.0$
 A  AV  DO3 I=1,3$
        AXGR(I)=X(I)*D GR-(XDGR(I)*CAPD+XDOT(I)*DGR) $
     3  E2VGR=A GR(I)*EQX(I)+E2VGR$
        XL  =E  GR/XLGR$
        XL   =  (-D   /RTA)*2.0+XLGR+SMLGR $
 H  AV  HX  =(Z GR* -Y GR*Z)/SQRMU$
        HYGR=(X GR*Z-Z GR*X)/SQRMU$
        HZGR=(Y GR*X-X GR*Y)/SQRMU$
```

```
*     C  PUTE R  LIGHT  ERIVATIVES
 C  VT  RKS C(   )=XLGR*XKE+?? $
        DO  I=1,3$
        RKS  (  +I)=AY  (I)*XKE$
         (  TO(  ?+I) = RXG (I) * YNR$
        ST  T  T  C$
   ...  U      (R)"
        PA-      $
*       CO TROL ROUTINE FOR EPHEMERIS INTEGRATION
*       CALLED  T COMPLETION  OF EACH INTEGRATION STEP
        EN TAC     
*   IF  IATELY AFTER EACH PERIGEE PASSAGE, CORRECT ORBIT PARAMETERS
*           FOR  TMOSPHERIC DRAG (RECTIFY ORBIT)
 C  TRL   IF(SSI  )32,33,33$
      32RECTI=0.0$
        GO TO C  TRL2$
 3      IF(RECTI)CNTRL2,34,CNTRL2 $
      34RECTI=1.0$
        CALL RCTFY(NMPER,ESINE,ECOSE,A,WX,URGCO,RKSTO,XN,XNZ,AYNI,
        U7, )   $
        START  AC $
        U P      DERIV$
        END TAC   $
 C TRL2 EPHEM(NCNTRL)=RKSTO(1)$ CURRENT TIME (SINCE EPOCH) TO EPHEM BUFF
 S  MP2  IF(NCNTRL-IXXX) 35,OFLO,OFLO$ IF EPHEM BUFFER IS FULL,GO WRITE
 35     DO 36 I=1,7$
        NCNTRL= CNTRL+1$
*     T  NSFER CURRENT VALUES OF XL, ABAR, H BAR TO EPHEMERIS BUFFER
 3       EPHEM( CNTRL) = RKSTO(I+4)$
        COU T = COU T + 1 $
        NCNTRL = NCNTRL + 1 $
          IF(SE SE LIGHT 10)  X36, X36$       TURN OFF SENSE LIGHT 10
 X3     IF(CNT  (1))38,SEC5E,3 $
*        DETERMINE WHAT TO OUTPUT FOR PRINTING
 3      IF(IPNTFL-1)LI E7,41,SEC5E$
 SEC5E   IF(SENSE LIGHT 10)43,X44$      PRINT R BAR, R BAR DOT
   44   WRITE T PE 6,RKSTO(1), (X(I),I=1,3), (XDOT(I), I=1,3)$
        NTAPE6= TAPE6+1 $
        SENSE LIGHT 10$
        IF(SENSE SWITCH 35) 36,38  $
 X4     CONTINUE $
        DO (XEL ) I=1,7 $
 X  L2  XEL1(I)=RKSTO(I+4)$
        CALL X  (XEL1,  XS,EL1,EL2,EL3,EL4,EL5,EL6)$
        WRITE T PE 15,RKSTO(1),EL1,EL2,EL3,EL4,EL5,EL6 $
        NTAPE15=NTAPE15+1   $
        GO TO 3 $
 4      IF(IPNTFL-2)LI E7,41,41$
 41      IF(1.-U7**2)45,46,45$ PREPARE LAT-LONG,ALTITUDE, FOR PRINTOUT
        XL   = SIGN  ( 0.,  Z) GO TO 49$
 4      XLAT=U  / (SQRT(1.-U7*U7)*0.99330658 ) $
        XLAT=ATAN (XLAT)*RADYN$
        IF( (1)) E( .), XLO G = SIGNF(90.,Y)@ GO TO 56 $
      4 XLO G=  AT (Y/ )*RADYN$
        IF (X( )) L  ( .),   LO G = XLONG + 180. $
      5 XLO G=  ONG+(R STO-TC)*ELP2F-THGR$
        XL     = 360.0 -APROPR(XLONG) $
        H=  LE( Z, , X  R)$
        IF(LI  -54)61,61,62 $
 6      CALL  R  (I  C T)$
         LI  =        I  C  =I  C T+1      $
```

```
      1LINE=LINE+1$
        CALL PRINT(RKSTO(1),ALT,XLONG,H)$
 LINE7 IF((RKSTO-TF)* T)GT=(0.),GO TO CSEND$     JMP IF THRU INTEGRATING
*         IS THERE   RESTART TIME (NEW EPOCH)
 71      IF(CNT D(10))     72, HELL, 72$
*     LOGIC TO STORE VALUES OF PARAMETERS WHEN RESTART TIME REACHED
     72IF(CNT  (13))IRENE,74,IRENE $
     74IF(CNT   (12))75,76,75$
        START TAC $
 HELL   JMP       RUNGE,RETURN$                  RESUME INTEGRATION
        END TAC    $
 76      IF((RKSTO+RKSTO(4)-CNTWD(10))*DT)LT(0.), GO TO HELL$
        RKSTO(4)=CNTWD(10)-RKSTO@CNTWD(12)=RKSTO(4)$
         GO TO HELL$
     75IF(CNTWD(11))80,81,80$
     80CNTWD(13)=CNTWD(11)$
        GO TO 82$
     81RKSTO(4)=DT-RKSTO(4)@CNTWD(13)=RKSTO(4)$
 82     CNTWD(14) = PROPR (RKSTO(5)) $
     84DO87I=1,3$
 87     CNTWD (I+16) = HXI (I) $
        CNTWD(15) =AXNI(1)@ CNTWD(16)= AYNI $
         GO TO HELL$
 IRENE   IF(CNTWD(11)) E (0.), RKSTO(4) = DT$
         GO TO HELL$
*   WRITE EPHEM BUFFER  ON TAPE 3, TURN ON SENSE LIGHT 4
 OFLO   WRITETAPE3,(EPHEM(1),I=1,IXXX) $
        NTAPE3=NTAPE3+1 $
        SENSE LIGHT 4@ NCNTRL = 1@ GO TO CNTRL2 $
        START TAC$
        PAGE     $
*      CSEND RECEIVES CONTROL AFTER INTEGRATION OF EPHEMERIS
*  IF REQUESTED ,OUTPUT CODE 2 OR 3, PRINTS X,Y,Z EPHEMERIS
* RUNS OUT EPHEM BUFFER ONTO TAPE 3
*   SENDS CONTROL  TO DIFFERENTIAL CORRECTION OR SIMULATION
        END TAC $
 CSEND  IF(IPNTFL)LT(2),GO TO DOBSM1 $
        REWIND 6 $  YES - PRINT X,Y,Z EPHEMERIS, FROM DATA ON TAPE 6
        IPGCNT = 1$
        DO 225 I=1,15$
 225    CHED(I) = CHED2(I) $ TIME X Y Z XDOT YDOT ZDOT  - TO HEADING
        LINC=60  $
        ITEMP=NTAPE6 $
 RDTAP  READ TAPE 6,(OTPT(I),I=1,7)$
        NTAPE6= TAPE6-1 $
 226    IF(LINC-54)227,227,228 $
 228    CALL HEAD (IPGCNT)$
        LINC=5@IPGCNT=IPGCNT+1$
 227    LINC=LINC+1$
        WRITE OUTPUT TAPE 5,2227,(OTPT(I),I=1,7)$
 2227   FORMAT(2H1 , 14X, 7F15.8)$
 3227 IF( TAPE6) GT (0), GO TO RDTAP$   LOOP TILL ALL PRINTED
        NTAPE6 = ITEMP $
 DOBSM1 IF(SENSE SWITCH 35)C36,DOBS  $
 C35    CONTINUE        $
        REWIND  15 $
        IPGCNT=1 $
        LINC=60 $
        ITEMP=NTAPE15 $
 C36    READ TAPE 15,(OTPT(I),I=1,7)$
        NTAPE15=NTAPE15-1 $
```

```
       IF(LIN -64)F36,F36,F36 $
F3     WRITE   TPUT TAPE 5,ELEM1  $
ELEM1  FO  AT( F7 ,37HKEPLERIAN ELEMENTS FOR EACH TIME STEP///)$
       WRITE   TPUT TAPE 5,ELEM2 $
ELEM2  FO  AT( H10,6X,4HTI E,10X,4HAXIS,11X,5HECCE  ,9X,6HINCLIN,9X,
       6H  ANG ,8X,8HARG PERG,7X,8HR A NO E//)$
       LI  =5  $
       IPGCNT=IPGC T+1  $
       LI C=LI C+1
       WRITE   TPUT TAPE 5,G36,(CTPT(I),I=1,7)$
G3     FO  AT(FH1 ,7F15.8)$
       IF( TAP 15)ST( ),GO TO F36 $
       NTAPE15=ITE P  $
       RE IND 15$
D  SM  SENSE LIGHT 4REWIND 6$
       IF( TAPE3)D BS 3,DOBSM3,DOBSM4$
DOBSM3 IXXX=   NCNTRL$
* RUN OUT EPHE  BUFFER ONTO TAPE 3
D  SM4 WRITE T PE 3,(EPHEM(I),I=1,IXXX ) $
       NTAPE3= TAPE3+1 $
       NTAPE3X=NTAPE3  $
CNTRD  IF(SENS  LIGHT 2)  9, 90$ ARE OPS ON TAPE 9
  9    SENSE LIGHT 2 $                     YES - GO DO CORRECTION
       GO TO  RBCO$
 90    IF (CNT D(1))    ORBCO, 92, ORBCO  $  SIMULATION OR CORRECTION
       STA T 1 C$
       PAGE     $
*  SI ULATION  UN
       SAVE    TMOTO, RCARD,T OTO $
       END TAC $
 92    LINE =  00       IPGCNT=1 $
       ISAVE6= TAPE6$
       DO  3 I=1,15$
     93CHE (I)=OCHED(I)$
       CASE=CASE2$
  SIM  CALL RE DI(CNT D,STARC,IHED,NCARD)$
R  DI  GO TO(   ,N STA),NCARD$
 0     PHI D=STARC(3)*CONV$
       XL  BA = APPOP  (-STARC(4)) $
 9     SI PH(1)=SI (P IRD) $
       CAPC=(1.0-EPSQ *SI PH**2)$
       CAPC=1. /SQRT (CAPC)$
       CAPS=CAPC*(1.0-EPSQ )$
       OALT=ST RC(5)/  PER$
       CAP =(  LT+CAPS)*(-SINPH)$
       COSPH=C S(PHIR )$
       XOVCT=( ALT+CAPC)*(-COSPH)$
 1     IF( TAP 6)100,100,101 $
 100   RE IND  $
       NTA E6= SAVE6$
       GO TO   SI
 101   RE  TAPE6,(TXY (I),I=1,4),(OBSCD(I),I=1,3)$
       NT  E6= TAP 6-1 $
CH HC  XL   THE=(TH  +XLAMBA)*C  V $
       IT  P=( RG  +TXYZ)/X  NPEA$
       T= TE   
       IT  P=  E  +IT $
       T=(  RG  +TAYZ)/X   E A-T $
       EQ =(1.-T)*EQE (IT  P)+T*EQEG(ITEMP+1) $
       EQ =EQ -CO  /2   .$
       THT =(T YZ-T )* PT1  +Y  PTH+EQQ  $
```

```
          COSTH=  S(T TA)$
          CA  =X   CT*COSTH$
          RH  =C   X+T YZ(2)$
          SI TH=SIN(THETA)$
          CAPY=X  CT*SINTH$
          RHOY=CA Y+T YZ(3)$
          RHOZ=CA Z+T YZ(4)$
          RHO1=S  T(RHOX**2+RHOY**2+RHOZ**2)$
      1 5XLX=RH  *COSPH
          XLZ=RH  *SI TH+RHOX*COSTH$
          XLX=(XLZ*SI PH-XLX)/RHO1$
          XLY=(COSTH*RHOY-SINTH*RHOX)/RHO1$
          XLZ=(XLZ*COSPH+SINPH*RHOZ)/RHO1$
          IF(XLZ)104,106,106$
 106      IF(XLX(1))COMPA,107,COMPA$
      107IF(XLY)108,ERROR,109$
      108TEMP=3.0*PI/V2$
          GO TO C VDG2$
      109TEMP=PI V2$
          GO TO C VDG2$
 COMPA    CAPA=ATAN(XLY/(-1.0*XLX))$
          IF(XLX(1))110,110,111$
      110IF(XLY)112,113,113$
      111TEMP=CAPA+PI$
          GO TO C VDG2$
      112TEMP=CAPA+2.0*PI$
          GO TO C VDG2$
      113TEMP=CAPA$
 C VDG2   CAPA=TE P/CONV$
          TEMP=1.0-XLZ**2$
          IF(TEMP)LTE(0.),SMLF=SIGNF(90.,XLZ)@ GO TO H2DEG $
          SMLF = ATAN(XLZ/SQRT(TEMP))/CONV $
 H2DEG    TEMP = (ORGTM + TXYZ) / XMNPDA $
          ITE P=TEMP$
          CNTWD(14) = ITEMP+ ORGDA$
          CNT D(15) = TEMP - ITEMP$
          CNTWD(15) = CNTWD(15)*XMNPDA$
*A PUNCHROUTINE MUST BE ENTERED HERE TO PUNCH OUT$
*AN OBSVCARD. USING CASE NO AS SATELITE NO.$
*         STATION NO FROM CNTWD( ) IN BCD,CURRENT YR$
*         DAY NO FROM CNTWD(14), TO BE CHANGED TO MONTH$
*         AND DAY AND FROM CNTWD(15) TO HRS, MIN AND SEC.$
*         $
*         $
          DO 300  =2,13$
          ITE P1=I-1 $
          IF(TMOT (I)-CNTWD(14))300,300,301$
 300      CONTINUE$
 301      ITE P2=CNTWD(14)-TMOTO(ITEMP1) $
          ITEMP3=(CNT D(15)/60.0)$
          TEMP2=(CNTW (15)-(ITEMP3*60.0))$
          ITE P4=TEMP2$
          ITE P4=TEMP2+.0001 $
          TEMP2=(TEMP2-ITEMP4)*60. $          SECONDS AND FRACTIONS
          IF(TEMP2) LT(0.0), TEMP2=0.0 $
          INT P2=TEMP2 $                      SECO DS
          INT P3=((TE P2-INT P2)*10000.)+.00000001 $
          STA T T C $
          TMA      INT P2 $
          SRA      32 $
          TA       INT P2 $
```

```
S         BI  RC  INT P2 $
          TA       IlT P2 $
          T        INT P3 $
          SR       3$
          TA       INT P3 $
S         HI  FC   INT P3 $
          TA       INT P3 $
          E   TA  $
          TE  2=(TEMP -I  MP4)*6 0000.0$
          TE  =S  H*1 00 .0$
          TE  3=C PA*100  .0$
          TE  4=R 01*  MP R$
          DO(X905)I=1,3$
X 05      XLS BX(I)=RHOX(I)/RHO1$
          TE  5=1.0-XLSUBZ**2$
          IF(TEMP5)X914,X914,X915$
X 14      IF(XLSU Z)X916,X917,X917$
X 16      TEMP5=-1.0*PIOV2$
          GO TO  LTA2 $
X 17      TEMP5=PIOV2$
          GO TO  LTA2D$
X 15      DELTA=SQRT(TEMP5)$
          TE P5=ATAN(XLSUBZ/DELTA)$
DLTA2     DELTA=TEMP5/CONV$
          IF(XLSUBX(1))CALFA,X907,CALFA$
X  7      IF(XLSU Y)X908,ERRO ,X909$
X  8      TEMP5=3.0*PIOV2$
          GO TO ALF2H$
X9 9      TE P5=PIOV2$
          GO TO ALF2H$
CALFA     ALPHA=ATAN(XLSUBY/XLSUBX)$
          IF(XLSU X(1))X911,X911,X910$
X 10      IF(XLSU Y)X912,X913,X913$
X 11      TEMP5=ALPHA+PI
          GO TO ALF2H$
X 12      TE P5=ALPHA+2. *PI$
          GO TO ALF2H$
X 13      TEMP5=A PHA$
ALF2H     ALPHA=TEMP5/(CONV*15.0)$
          CPX T=-1.0*CAPY*THDOT$
          CPYL T=C PX*THDOT$
          RRATE=XLSUBX*(OBSCD+CPXDT)+XLSUBY*(OBSCD(2)+CPYDT)$
          RRATE=RRATE+XLSUBZ*OBSCD(3)$
1         IF(LINE-54)119,119,120 $
12        CALL HEAD (IPGCNT ) $
          LINE=5  IPGCNT=IPGCNT+1 $
11        LINE= LINE +1  $
          ALFA=ALPHA*15. $
          WRITE OUTPUT TAPE 5,P1,CNTWD(8),TXYZ(1),RHO1,RRATE,ALFA,
          DELTA,CAPA,SMLH $
P1        FORMAT(2H10,A4,7X,7F15.8) $
*         SENSE S ITCH 30    PUNCH AZ AND EL  CARDS
*         SE SE S ITCH 31    PUNCH RA AND DEC CARDS
*         SENSE S ITCH 32    PUNCH RANGE CARDS
*         SENSE S ITCH 3     PUNCH RANGE RATE CARDS
          IF (SE SE S.ITCH 31) 7300,7301 $
7         INRAH=ALPHA $                   RT. ASCENSION TO FIXED POI T
          RAFRK=I RAH $                   HOURS TO FLOATING POINT
          RA  NE=(ALPHA-RAFRK)*60. $      FRACTIONAL HOURS TO MINUTES
          INRAN=  MIN $                   TO FIXED POINT
          RAFRN=I RA  $                   BACK TO FLOATING
```

```
          RASEC=(RAMIN-RAFRK)*600. $       SECONDS/10
          INRAS=RASEC $                    CONVERT TO FIXED POINT
          DEETA=DELTA $                    ASSUME DECLINATION TO
          DELSN=PLUS $                      BE POSITIVE ==>
          IF(DELTA) 7302,7303,7303 $       UNTIL PROVEN OTHERWISE --
 7302     DELSN=XMINS $
          DEETA=-DELTA $                     CHANGE SIGN OF DECLINATION
 7303     INDCD=DEETA $                    DECLINATION DEGREES TO FIXED POINT
          DCFRK=INDCD $                    INTEGER TO FLOATING POINT
          DCMIN=(DEETA-DCFRK)*60. $        FRACTIONAL DEGREES TO MINUTES
          INDCM=DCMIN $                    MINUTES TO FIXED POINT
          DCFRK=INDCM $                    BACK TO FLOATING
          DCSEC=(DCMIN-DCFRK)*600. $       SECONDS/10
          INDCS=DCSEC $                    TO FIXED POINT
          START TAC $
          TMA       INRAH $
          SRA       32 $
          TAM       INRAH $
S         BIN2BCD   INRAH $
          TAM       INRAH $
          TMA       INRAM $
          SRA       32 $
          TAM       INRAM $
S         BIN2BCD   INRAM $
          TAM       INRAM $
          TMA       INRAS $
          SRA       32 $
          TAM       INRAS $
S         BIN2BCD   INRAS $
          TAM       INRAS $
          TMA       INDCD $
          SRA       32 $
          TAM       INDCD $
S         BIN2BCD   INDCD $
          TAM       INDCD $
          TMA       INDCM $
          SRA       32 $
          TAM       INDCM $
S         BIN2BCD   INDCM $
          TAM       INDCM $
          TMA       INDCS $
          SRA       32 $
          TAM       INDCS $
S         BIN2BCD   INDCS $
          TAM       INDCS $
          END TAC $
          PUNCH (OBSCR),JXYR,CNTWD(8),IXYR,ITEMP1,ITEMP2,
          ITEMP3,ITEMP4,INTMP2,INTMP3,INRAH,INRAM,INRAS,DELSN,INDCD,INDCM,
          INDCS $
 OBSCR    FORMAT(I2,3X,1H2,4X,A4,I1,4I2,W2,W4,1H-,2W2,W4,W1,W3,W2,W3 ) $
 7301     IF (SENSE SWITCH 30) 7400,7401 $
          START TAC $
 PLUS     A/         $
 XMINS    A/        -  $
          END TAC $
 7400     INAZD=CAPA $
          AZFRK=INAZD $
          AZMIN=(CAPA-AZFRK)*60. $
          INAZM=AZMIN $
          AZFRK=INAZM $
          AZSEC=(AZMIN-AZFRK)*6000. $
```

```
          INAZS=AZSEC $
          INELD=ELH $
          ELFRK=IELM $
          ELMIN=(SMLH-ELFRK)*60. $
          INELM=ELMIN $
          ELFRK=IELM $
          ELSEC=(ELMIN-ELFRK)*600. $
          INELS=ELSEC $
          START   TAC $
          TMA     INAZD $
          SRA     32 $
          TAM     INAZD $
S         BIN2BCD INAZD $
          TAM     INAZD $
          TMA     INAZM $
          SRA     32 $
          TAM     INAZM $
S         BIN2BCD INAZM $
          TAM     INAZM $
          TMA     INAZS $
          SRA     32 $
          TAM     INAZS $
S         BIN2BCD INAZS $
          TAM     INAZS $
          TMA     INELD $
          SRA     32 $
          TAM     INELD $
S         BIN2BCD INELD $
          TAM     INELD $
          TMA     INELM $
          SRA     32 $
          TAM     INELM $
S         BIN2BCD INELM $
          TAM     INELM $
          TMA     INELS $
          SRA     32 $
          TAM     INELS $
S         BIN2BCD INELS $
          TAM     INELS $
          END TAC $
          PUNCH(OSCAR),IXYR,CNTWD(8),IXYR,ITEMP1,ITEMP2,
          ITEMP3,ITEMP4,INTMP2,INTMP3,INAZD,INAZM,INAZS,INELD,INELM,INELS$
  OSCAR   FORMAT(I2,3X,1H2,4X,A4,I1,4I2,W2,W4,W3,W2,2W4,W2,W3)$
  7401    IF (SENSE SWITCH 32) 7500,7501 $
  7500    EXPUT=1.0 $
          DO 7550 I=1,99 $
          EXPUT=10.*EXPUT $
          IF(TEMP4) LT (EXPUT), GO TO 7560 $
  7550    CONTINUE $
  7560    IJTT=I $
          START TAC $
          SRA     32 $
          TAM     IJTT $
S         BIN2BCD IJTT $
          TAM     IJTT $
          END TAC $
          TEMP4=TEMP4/EXPUT $
          PUNCH(OSCAR),IXYR,CNTWD(8),IXYR,ITEMP1,ITEMP2,ITEMP3,ITEMP4,
          INTMP2,INTMP3,TEMP4,IJTT $
  OSCAR   FORMAT(I2,3X,1H1,4X,A4,I1,4I2,W2,W4,5X,F11.11,1H+,W2)$
  7501    IF(SENSE SWITCH 33) 7600,7601  $
```

```
7600    RRJTT=    .1*((F.*37922.43-100.*RRATE)/(37922.43+RRATE))$
        IS=10$
        PUNCH (OSSCR),IXYR,CNTMP(8),IXYR,ITEMP1,ITEMP2,
        ITEMP3,ITEMP4,INTMP2,INTMP3,IS,RRJTT  $
OSSCR   FORMAT(I2,3X,1H3,4X,A4,I1,4I2,W2,W4,8X,I2,F9.9) $
7601    GO TO 1 4$
         START TAC$
        PAGE$
*        DIFFERENTIAL CORRECTION RUN
        END TAC   $
OHECO   REWIND9 $
        IOBCNT2=IOBCNT  $
        IF (SENSE LIGHT 4)121,122$
121     SENSE LIGHT4$
        IF (EFLAG)123,123,122$
  123   REWIND 3 $  IF EPHEMERIS ON TAPE 3,   REWIND 3
122     NTAPE3= NTAPE3x $
        IF (SENSE LIGHT 3) 124, 124 $     TURN OFF SENSE LIGHT 3
 124     IPGCNT = 1@   SUM = 0.0@ SUM2 = 0.0$
         IF (SENSE LIGHT 2) INITL,125$ IF FIRST PASS THRU OBS, RD FM CRD
*        READ STATION AND OBSERVATION CARDS
125      IF (IRPT)GT (9), IRPT=0$   SET NO. OF TIMES TO REPEAT CORR.
RREC    CALL READER(STARC,ISTCNT,LSTCNT,IASIZE,IDBUF,XMPER)$
         START TAC         $
        TMA       LSTCNT               $
        JMP      CHGNXN $      SET UP LEAST SQUARES ROUTINE FOR MATRIX SIZ
        PAGE$
        END TAC  $
*     READ OBSERVATION CARDS, STORE ON TAPE 9
        IALL=1@IOBCNT=0$
*ENTER  RCARD AND READ 1 OBSV CARD STORING$
*UTIME, ALPHA,DELTA,RANGE, AND STAID = STA NO IN BCD$
*WHEN   LAST CARD IS READ SET IALL=2$
X130    CALL RCARD(SATEL,STAID,ASIG1,IBMUTH, DAYNO, ALPHA,
        DELTA,RANGE,RODTO,UTIME,NCARD,XMPER)$
        GO TO (X132, 131), NCARD $    CHECK IF ALL OBS. CARDS READ
  131    REWIND 9$    ALL OBS. CARDS READ - GO DO CORRECTION
        GO TO INITL$
 X132   I = 1 $    NO. OF STATION RECORD IN STARC
        START TAC $
        TMA      ISTCNT$
        SLA      3$
        AD       $
        AM       C/HLT,STARC$
        TDXLC    ,3$
        TAM      JXXY$  JXXY HAS ADDRESS OF END OF STARC TABLE
*   FIND STATION IN STARC BUFFER
ZXXY    TMA      STAID$
        TMD      ,3$
        JAED     H134$   SEARCH FOR STAID IN STARC TABLE
        TMA      N/9T15$
        AMS      I $
        TMD      JXXY$
        AIXOL    9,3 $
        JNO      ZXXY$
        TMD      STAID$    STATION NOT FOUND - NOTE ON FLEXO
        TDM      BLAB3+2$
        TMA       C/HLT,BLAB3@ C/HLT,5$
        JMPL     6$           AND ON OUTPUT TAPE 5
        END TAC$
        WRITE OUTPUT TAPE 5, 705, STAID@  GO TO X130$
```

```
7¹5     FORMAT(10H10STATION  ,A8, 10H NOT FOUND   )$
-134   PHI D = STARC(I+1) $    GET LATITUDE, EAST LONGITUDE, HEIGHT
       XLA BA = STARC(I+2) & OALT = STARC(I+3) $
        STARK = STARC(I+4)$                  GET IAXD
       DO 708 J = 1,4 $
70     IF(ASIG1(J)) E (0.), ASIG1(J) = STARC(I+J+4) $
       XMN TS = UTIME -ORGTM + XMNPDA*(DAYNO-ORGDA)$  MINUTES SINCE EPC
       TEMP = XMNUTS - TO $
       IF(TEMP*DT) LT(0.), GO TO 706$     FIND IF THIS TIME IN EPHEMERS
       IF(TEMP *(TF-XMNUTS)) GTE (0.), GO TO 707$
706    WRITE OUTPUT TAPE 5, 709, STAID, DAYNO$   GO TO X130 $
709    FORMAT (19H10OBSERVATION FROM ,A8,3H ON,F4.0,18H OUT OF TIME RAN
       GE ) $
707    SINPH = SIN (PHIRD) $
       TEMP=(1.0-SINPH**2*EPSQD)$
       CAPC=(1.0/SQRT (TEMP))$
       CAPS=CAPC*(1.0-EPSQD)  $
        CAPZ=(CAPS+OALT)*(-SINPH)$
       COSPH=COS (PHIRD)$
       XOVCT=(OALT+CAPC)*(-COSPH)$
        XLMPTH = THGRL + XLAMBA       $
       IDAY=DAYNO-ORGDA+IT  $
       IITT=UTIME/1440. $
       T=UTIME/1440.-T       $
       T=IITT$
       EQQ=(1.-T)*EQEQ(IDAY)+T*EQEQ(IDAY+1) $
       EQQ=EQQ*CONV/240.$
       THTA=XMNUTS*RPTIM+XLMPTH+EQQ $
       COSTH=COS (THTA)$
       CAPX=XOVCT*COSTH$
       SINTH=SIN (THTA)$
       CLM=COS(THTA-XLAMBA)$
       SLM=SIN(THTA-XLAMBA)$
       CAPY=XOVCT*SINTH$
153    CALL  GTOBS(IOTYPE)$      COMPUTE L BAR, ETC.
       IF(SENSE LIGHT 5)  X130, 151$ GTOBS TURNS THIS ON WHEN OBS DONE
151    WRITETAPE9,SATEL,STAID,IOTYPE,XMNUTS,
       (CAPX(I),I=1,3),(ASUBX(I),I=1,9),ASIG1,IAXD,SPHA, CPHA, SDEL,
       CCDEL,CLM,SLM,IBMUTH,EQQ $
       IOBCNT=IOBCNT+1$
        IOBNT2=IOBCNT $
       IF(IOTYPE)152,152,X151$
   X151IF(IOTYPE-2)X152,X152,X130$
   X152ALPHA=0.0@DELTA=0.0$
       GO TO 153$
152    RANGE=0.0$
       GO TO 153$
       START TAC$
       PAGE$
*   OBSERVATIONS ARE ON TAPE 9 - MAKE PASS THRU THEM.
       END TAC $
INITL  SENSE LIGHT 2$
       IF(SENSE LIGHT 4)154,155$
154    READ TAPE3,(EPHEM(I),I=1,IXXX) $
       SENSE LIGHT 4  $
       NTAPE3= TAPE3-1 $
155    RCNT=0.0@RCNT2=0.0$
       BOCNT=0.0$
       LSTEP=1 EFC T=0.0$
       ITSTAC=  $
       IRC T=  IPC T2=0$
```

```
*         ENTER LEAST SQUARES HE F$
          ST  TT         $
                LSG          $
           E  TAL    $
* PASS THRU OBSERVATIO S
 FU CT   IF(SFNS  LIGHT3)178,179$
*    JUST DI  RA OR AZI UTH - DO DECL OR ELEVATION
 178      ARS L=RESID$
          DO (X1  ) I = 1,3 $
 X1 0     ARAY(I)=WST (10+I) $
          CALL COEFF(ARAY,I,X PER)$
          GO TO (180,RADCB,OMIT1),I             $
 180      IF(IWST -2)A180,A180,R180             $
 A180     SU  = RESID**2 + SUM $            FORM SUM OF SQUARES OF RESIDUALS
          RCNT=RC T+1.0   $
          IRCNT=RCNT       $
          GO TO F NEX     $
 R180     SUM2=RESID**2+SUM2                    $
          RC T2=R NT2+1.0$
          IRC T2=RCNT2    $
 FI EX    GO TO LSQR$         ADD THESE COEFFICIENTS TO LEAST SQUARES MATRIX
 179      IF(RCNT+RCNT2)181,181,PROBXX          $
 181      DO 182 I=1,18                         $
  182     CHEC(I)=OBS D(I)$ SAT. STA. TIME RHO RESID/KM RA RESID/KM...ETC.
 1        CALL HE D(IPGCNT)                     $
          IPGCNT=IPGCNT+1                       $
          LINE=6                                $
          GO TO TSEOF                           $
 PROBXX IF(IWST  - 1) A189, B189, F189          $
 F189     IF(IWST ) E (3), APRNT13 = RESID@ GO TO PROBS$
          APR T71 = RESI    $
          APR T56=ARS L                  $
*         PRINT ACCORDING TO PAGE 59$
 PR BS   IF(LINE-54)PROBSX,PROBSX,PROBSY$
 P  BSY  CALL HE D (IPGCNT)$
         IPGCNT=IPGC T+1$
         LI  =5$
 P  BSX  LI  =LI E+1$
         WRITE   TPUT TAPE 5,CVTOBJ,WSTO(1),WSTO(2),WSTO(4),APRNT41,
         APR T56,APR T71,APR T86,APRNTO1,APRNT13, APRNTS $
 C TOB   FOR AT(2H10,A3,3X,A4,7E14.7,2X,A4)$
 C TOP   APR T41 = 0.0 $
         AP  T56 = 0.0 $
         APR T71 = 0.0 $
         APR T86 = 0.0
         AP  TO1=0.0 $
         AP  T13= 0.0        $
         APR TS = APRNT  $
  TSEOF  IF(IOB T2)EFCH ,EFCH2,185$
     1   I   T2=IO  T2-1       $
         GO TO   VO
     T1  C   T=C CNT+1$
         AP  TS = ST RS
         IF (SE  E LIGHT 3) 178, PROBXX $
         AP  T4  =RESID  $
         GO    PROBS $
         AP  T   =A  DL
         AP  TO  = R SI   $
         GO    PROBS $
         IF( STO T-I CN -IRC T2)CTV1,CTV1,UNKER$
         ST  T   C$
```

```
UPPER  TMA      C/FLT,  BCBF C/FLT,6$
       UMPL     6$
       RU OUT    $
       UMPL     5$
       END TAC $
 CTV1  GO TO LSQS1$          GO FILL IN A MATRIX
       J = LSTCNT**2 $
       DO 720 K=1,J$
 720    SQMA(K) = QLSQA(K)$      SAVE A MATRIX IN SQMA
       DO 721 K = 1,LSTCNT $
 721    SQBA(K) = QLSQB(K)$            SAVE B MATRIX IN SQBA
          LMODE = 1$
       IF (MCODE (1)) 722, 723, 722 $
    722GO TO LSQS$
       WRITE OUTPUT TAPE 5,724$
 724   FORMAT(59H20CORRECTION MADE TO ORBITAL ELEMENTS AND STATION POSI
       TION.)$
       GO TO CHEKT$
       GO TO CUMBAC  $
*
 723       J = LSTCNT**2$
        DO 725 K = 1,J $
 725   SQMF(K) = SQMA(K)$         MATRIX A TO MATRIX A1
        DO 726 K = 1, LSTCNT   $
 726    SQMB(K) = SQBA(K)$            MATRIX B TO MATRIX B1
       L=6$
       K = L * LSTCNT $
       DO 727 M=1,K$
 727   SQMF(M)=0. $ SET 1ST L ROWS OF MATRIX A1 = 0
 728   K = L * LSTCNT + 1 $
       M = K + 5 $
       DO 729 N=K,M$
 729   SQMF(N) =0.$  SET 1ST L ELEMENTS OF REMAINING ROWS = 0.
       L=L+1$
       IF (L - LSTCNT ) 728, 730, 730 $
    730K=LSTCNT+1$
       L=6*LSTCNT-3*IASIZE$
       DO 731 M=1,L,K$
 731   SQMF(M) = 1.0$      SET FIRST L DIAGONAL ELEMENTS = 1.
       DO 732 M=1,6$
 732   SQMB(M) = 0.0$   SET FIRST L ELEMENTS OF MATRIX B1 = 0.
          LMODE = 2$
       IF (MCODE(2)) 733, 734, 733 $ SHALL WE CORRECT STATIONS ONLY
    733DO 735 M=1,J$
 735   QLSQA(M) = SQMF(M) $     YES-TRANSER A1 AND B1 MATRICES TO
       IF(IASIZE) 5 (U), WRITE OUTPUT TAPE 5, 773@ GO TO CUMBAC $
  773  FORMAT (2H10 / 38H10NO STATION POSITIONS TO BE CORRECTED ) $
       DO 736 M = 1, LSTCNT$     WORKING AREA USED BY LEAST SQUARES
  736    QLSQB(M) = SQMB(M)$           SUBROUTINE
        GO TO  SQS$          AND GO SOLVE EQUATIONS
        WRITE  OUTPUT TAPE 5, 749$
  749  FORMAT(43H2 CORRECTION MADE TO STATION POSITION ONLY.)$
       GO TO CHEKT$
       GO TO CUMBAC  $
*
  734  J = LSTCNT **2 $
       DO 738 K = 1,J $      MATRIX A TO MATRIX A2 (SQMC)
 738   SQMC(K) = SQMA(K)$
       DO 739 K = 1, LSTCNT $    MATRIX B TO MATRIX B2 (SQBC)
 739   SQBC(K) = SQBA(K)$
       IF(LSTCNT-6)12134,12134,12135$
```

```
UIFER  TMA      C/FLT,  BCBF C/FLT,6$
       JMPL     6$
       RU  LT     $
       JMPL     5$
       END TAC $
 CTV1  GO TO LSQS1$          GC FILL IN A MATRIX
       J = LSTCNT**2 $
       DO 720  =1,J$
 7 0    SQ A(K) = QLS A(K)$       SAVE A MATRIX IN SQMA
       DO 721   = 1,LSTCNT $
 721    SQ A(K) = QLS B(K)$            SAVE B MATRIX IN SQBA
          LMODE = 1$
       IF (MCODE ( )) 722, 723, 722 $
   722GO TO LSQS$
       WRITE OUTPUT TAPE 5,724$
 7 4   FORMAT(59H20CORRECTION MADE TO ORBITAL ELEMENTS AND STATION POSI
       TIO .)$
       GO TO C EKT$
       GO TO C WBAC  $
*
 723       J = LSTCNT**2$
        DO 725 K = 1,J $
 725   SQ F(K) = SQMA(K)$        MATRIX A TO MATRIX A1
        DO 726 K = 1, LSTC T  $
 726    SQ B(K) = SQBA(K)$            MATRIX B TO MATRIX B1
       L=0$
       K = L * LSTCNT $
       DO 727  =1,K$
 727   SQ F(M)=0.   SET 1ST L ROWS OF MATRIX A1 = 0
 7 8   K = L * LSTCNT + 1 $
       M = K + 5 $
       DO 729  =K, $
 729   SQ F(M) =0.$  SET 1ST L ELEMENTS OF REMAINING ROWS = 0.
       L=L+1$
       IF (L - LSTCNT ) 728, 730, 730 $
   73 K=LSTC T+1$
       L=6*LSTCNT-3*IASIZE$
       DO 731  =1,L,K$
 7 1   SQ F(M) = 1.0$      SET FIRST L DIAGONAL ELEMENTS = 1.
       DO 732  =1,6$
 73    SQFB(M) = 0.0$   SET FIRST L ELEMENTS OF MATRIX B1 = 0.
          LMODE = 2$
       IF (MCODE(2)) 733, 734, 733 $ SHALL WE CORRECT STATIONS ONLY
   733DO 735  =1,J$
 735   QLSQA(M) = SQMF(M) $     YES-TRANSER A1 AND B1 MATRICES TO
       IF(IASIZE) E (0), WRITE OUTPUT TAPE 5, 7736 GO TO CUMBAC $
  773  FOR AT (2H1  / 38H10NO STATION POSITIONS TO BE CORRECTED ) $
       DO 736    = 1, LSTCNT$      WORKING AREA USED BY LEAST SQUARES
 736     QLSQ (M) = SQFB(M)$            SUBROUTINE
        GO TO  SQS$            AND GO SOLVE EQUATIONS
        WRITE  UTPUT TAPE 5, 749$
 749   FORMAT(43H2 CORRECTION MADE TO STATION POSITION ONLY.)$
       GO TO C FKT$
       GO TO C WBAC  $
*
  734  J = LSTCNT **2 $
       DO 738   = 1,J $      MATRIX A TO MATRIX A2 (SQ C)
 7     SQ C(K) = SQMA(K)$
       DO 739   = 1, LSTCNT $    MATRIX B TO MATRIX B2 (SQBC)
 7     SQBC(K) = SQBA(K)$
       IF( STC T-6)12134,12134,12135$
```

```
12135   M=7-
        N = LST NT $
        K = 6 * LSTCNT $
   74   DO 741 L=M,J$
  741   SQMC(L) = 0.0 $   CLEAR FIRST SIX ROWS MATRIX A2, EXCPT 1ST 6 CL
        M=M+LSTCNT$
        N=N+LSTCNT $
        IF(N-K)740,740,742$
   742M=K+1$
        DO 743 L=M,J$
  743   SQMC(L) = 0.0 $ CLEAR REMAINING ROWS OF MATRIX A2
        M = K + 7 $
        K = LSTCNT + 1 $
        DO 744 L=M,J,K$
  744   SQMC(L) = 1.0 $ SET DIAGONAL ELEMENTS AFTER FIRST 6 TO 1.0
        DO 745 L = 7, LSTCNT $
  745   SQMC(L) = 0.0 $   CLEAR MATRIX B AFTER FIRST 6 ELEMENTS
12134   LMODE= 3 $
        IF (MCODE(3)) 746, 750 , 746 $ ARE WE TO CORRECT ORB ELEMS ONLY
   746DO 747 M=1,J$
747     QLSQA(M) = SQMC(M)$
        DO 748 M = 1, LSTCNT $ YES - TRANSFER A2, B2 MATRICES TO
  748   QLSQB(M) = SQBC(M) $   WORKING AREA USED BY LEAST SQUARES
        GO TO LSQS $      AND GO SOLVE EQUATIONS
         WRITE OUTPUT TAPE 5, 737$
  737   FORMAT(43H20CORRECTION MADE TO ORBITAL ELEMENTS ONLY.)$
        GO TO CREKTS
   750GO TO CUMBAC$
MOVOB   READ TAPE 9, (WSTO(I), I = 1,16), ASIG1, IAXD ,
        SPHA,CPHA,SDEL,CCDEL,COSTH,SINTH,IBMUTH,EQQ$
           I = IXXX$
  195   IF (NTAPE3) LTE (0), I = NCNTRL $
*       FIND OBSERVATION TIME IN EPHEMERIS
        IF(WSTO(4)-PASIEKA)RW1,RW2,RW2 $
RW1     REWIND 3 $
        READ TAPE 3,(EPHEM(J),J=1,IXXX)$
        NTAPE3=NTAPE3X-1 $
        LSTAD=1$ITSTAD=0 $
RW2     PASIEKA=WSTO(4)   $
          DO 198 J = LSTAD,I,8$
         LSTAD =J $
        IF(DT)CTV2,CTV3,CTV3 $
CTV3    IF(WSTO(4)-EPHEM(J))203,203,198 $
CTV2    IF(EPHEM(J)-WSTO(4))203,203,198 $
 198    CONTINUE       $
        IF(NTAPE3-1)TSTER,X201,201 $
X201    I=NCNTRL$
201     DO 202 J=1,24$
        KK = IXXX - 24 + J  $
 202     XXXX(J) = EPHEM(KK)$
        READ TAPE 3,(EPHEM(J),J=1,I)     $
        NTAPE3= TAPE3-1 $
        ITSTAD=ITSTAD+1  $
         LSTAD=1$
        GO TO 195$
*        OBSERVATION TIME FOUND
   203   IF(LSTAD-25)204,206,206$
   204    IF(ITSTAD)205,205,206       $
   205   J=1 $
         GO TO 209$
   206    J=LSTAD-24     $
```

```
208    IF(J+48-I ) 209,209,210 $
210    J=J-16  $
209    JJ=J+24 $
       DO 211  =1,8$
       TIMTP(K)=XXXX(JJ)$
211       JJ = JJ + 8$
*    INTERPOLATE TO GET PARAMETERS FM EPHEMERIS
CSFLP  DO 212  =1,6$
       CSUBI(K)=1.0&KK=K$
       DO 213 JJ=1,6$
       TEMP=TIMTP(KK)-TIMTP(JJ)$
       IF(TEMP)214,213,214$
214    CSUBI(KK)=CSUBI(KK)*(WSTO(4)-TIMTP(JJ))
       /TEMP$
213    CONTINUE$
212    CONTINUE$
FCNS   DO 215  =1,7$
       CSUM=0.0&KK=K+J$
       DO 216 JJ=1,6$
       CSUM=CSUM+CSUBI(JJ)*EPHEM(KK)$
 216   KK = KK + 8 $
       FJOFT(K)=CSUM$
 215   RKSTO (K+4) = CSUM $
       RKSTO(1)=WSTO(4)$
       CALL XYZSB $ GET X,Y,Z, RELATED PARAMETERS
       IF(TPFG)B1969,B1969,A1970 $
B1969  WRITE TAPE 6, RKSTO(1), X(1), Y, Z, XDOT(1), YDOT, ZDOT$
A1970  XNUZ=ZDOT              $
       A2OR=AR*AR&FSUBU=A2OR*ESINE$
       UMUO=(U-UO)*(-3.0/2.0)$
       RSUBA=(UMUO*RSUBU)+R$
       ROVA=R/A$
       RXN=(AXNI-COSEO)*A2OR$
       RYN=(AYNI-SINEO)*A2OR$
       USUBU=RTESQ*A2OR$
       USUBA=USUBU*UMUO$
       DENOM=XLGR**2*RTESQ$
       TEMP=(XLGR*ECOSE)*(-1.0)-ESQ+(1.0)$
       UYN=(TEMP/DENOM)*ESINE$
       UXN=UYN*AXNI$
       ROVA=1.0+ROVA$
       UXN=UXN+ROVA*SINEO$
       UXN=((-AYNI/XLGR)+UXN)*A2OR$
       UYN=UYN*AYNI-ROVA*COSEO$
       UYN=A2OR*((AXNI/XLGR)+UYN)$
       RHOC=0.0$
       DO217 K=1,3$
       RHOX(K)=WSTO(4+K)+X(K) $
 217   RHOC = RHOC + RHOX(K)**2$
       RHOC=SQRT (RHOC)$
       DO 218  =1,3$
       XLX(K)=RHOX(K)/RHOC$
 218    DELTX(K) = WSTO(K+13) - XLX(K)$
       IF(TPFG) A44, A44, B219 $
*    WRITE PARAMETERS FOR STD DEVIATION DETERMINATION
     A44WRITE TAPE 11,RKSTO(1),(RSUBA(I),I=1,4),(USUBA(I),I=1,4),
       (UX(I),I=1,3),(VX(I),I=1,3),R,COSI,SINI,SINU,COSU,
       (X(I),I=1,3),(DX(I),I=1,3)  $
*    INTERPRET OBSERVATION TYPE AND CALL COEFF TO COMPUTE
*      CORRECTION EQUATION COEFFICIENTS.
B219   IF(IVSI  ) 220,221,220          $
```

```
221     CALL COEFF(XLX,I,XMPER)$
         GO TO (180,BADOB,OMIT1),I   $
220     IF(1WST0-2)Y220,Y220,221$
Y220    SENSE LIGHT 3$
        DO(X220)I=1,3 $
X220    ARAY(I)=WSTO(7+I) $
        CALL COEFF(ARAY,I,XMPER)$
        GO TO(180,BADOB,OMIT1),I$
        START TAC  $
CHEKT   TJM      P22$
        END TAC $
         DO 219 I = 1,7$
 219     A11(I) = QLSQX(I)$
        IF(TPFG)X156,X157,X156$
 X157   TPFG = 5.0$  FIRST TIME HERE - PUT END SENTINELS ON TAPE 6,11
        WRITE TAPE 6,      HEND, X(1), Y, Z, XDOT(1), YDOT, ZDOT$
        REWIND  6       $
        WRITE TAPE 11,HEND,(STARC(I),I=1,25)$
        REWIND 11     $
X156     DO 156 I = 1,6 $
 156     DLTAA(I) = A11(I)$
        DO 158 I=1,18$
 158     CHED(I) = CHED3(I)     $
         CALL HEAD (IPGCNT)   $
        IPGCNT=IPGCNT+1$
        LINE=6 $
        IF(RCNT) X161,X161,X160 $
 X160   SUM11 = SQRT(SUM/RCNT)$  COMPUTE ROOT MEAN SQUARES OF RESIDUALS
X161    IF(RCNT2) X163,X163,X162 $
X162    SUM21=SQRT(SUM2/RCNT2)   $
*       OUTPUT CASE,SUM,DLTAA,OUTAP, SEE PAGE 61$
*              $
X163    WRITE OUTPUT TAPE 5,1011,ITIMES,IRPT,SUM11,SUM21,
        (DLTAA(I),I=1,6)$
1011    FORMAT(2H10,I1,2X,I1,F12.2,F12.8,6F15.8/2H70)$
        IF(IASIZE)XP1,P22,XP1$
XP1      WRITE OUTPUT TAPE 5, 1004$
1004    FORMAT(21H10SUGGESTED CHANGE IN )$
         WRITE OUTPUT TAPE 5, 1005$
1005    FORMAT(2H10,11X,29HCOORDINATES OF DATUM - METERS)$
        L=LSTCNT    $
        J=7$
        DO 809 K = 1,IASIZE $
        QLM(J)=QLSQX(J)*XMPER$
        QLM(J+1)=QLSQX(J+1)*XMPER$
        QLM(J+2)=QLSQX(J+2)*XMPER$
        WRITE OUTPUT TAPE 5,752,IDBUF(K),QLM(J),QLM(J+1),QLM(J+2)$
752     FORMAT(2H10,26X,I3,4X,10HDELTA X = ,F13.4,4X,10HDELTA Y = ,
        F13.4,4X,10HDELTA Z = ,F13.4)$
 809    J = J + 3 $
        START TAC $
P22     JMP      (P)$
        END TAC$
CUMBAC  IF((ABS(SUM11-GATE1)+ABS(SUM21-GATE2))*OMCNT*(OMCNT-OMCT2))
         161, CLACO, 161 $
161     OMCT2=OMCNT@OMCNT=0.0$
        GATE1 = SUM11 * X1STSG $ RESET GATES FOR ACCEPTANCE OF RESIDUALS
        GATE2 = SUM21 * X1STSG $
        IF(GATE1)A163,B163,A163$
 B163   GATE1 = SUM11 $
A163    IF(GATE2)162,C163,162$
```

```
 C163  GAT=2 = SUM21 $
1-2    EFLAG = 0.0        $
       IF (SE SE LIGHT 3 ) 164, 164 $  TUR  OFF SENSE LIGHT 3
1.     ITI ES=ITIMES-1$
       IF (ITI ES) CLACO, CLACO, ORBCO $   SHALL WE DO ANOTHER PASS
       GO TO F INT              $
CLACO  XLO=XLO+DLTLOG, XN=AXN+DLTAX   $
       AYN=AYN+DLTAY$
       A=(DLTAA+1.  )*A(2)$
       XNO E=DLTND+XNODE(2) $
       TEMP=X1 CL+DLTIN$
165    IF(TEMP)166,167,167$
166    TEMP=TE P+PI$
       GO TO 165$
167    XINCL=TEMP$
       COSI=COS (TEMP)$
       WZ=COSI$
       XLO = XLO + DLTND*SIGNF(1.0,WZ) $
170    RKSTO(5)=XLO$
       SINI=SIN (XINCL)$
       COSO=COS(XNODE(1))$
       SINO=SIN(XNODE(1))$
       WX=SINI*SINO$
       WY=(-COSO*SINI)$
       ESQ=AXN**2+AYN**2$
       P=(1.0-ESQ)*A$
       RTP=SQRT (P)$          P
       DO 171 I=1,3    $
 171    HX(I) = WX(I) * RTP @ HXI(I) = HX(I)$
       AXI=-COSI*AYN*SINO+COSO*AXN $
       AYI=(SINO*AXN+COSI*AYN*COSO)$
       AZI=AYN*SINI$
*
       X1A=A*AAXS $
       X2A=SQRT(ESQ)      $     ECCENTRICITY
       X3A=XINCL*RADYN    $     INCLINATION
       X3B=XNODE(1)*RADYN$
       X2B=ATAN (AYN/AXN(1)) $
       IF(AXN(1))X1X,X2X,X2X $
 X1X   X2B=X2B+PI$
 X2X   IF(X2B)X3X,X4X,X4X $
 X3X   X2B=X2B+TWOPI $
 X4X   X2B=X2B*RADYN $
       X1B=(XLO*RADYN)-X2B-X3B  $ MEAN ANOMALY
 X1C   IF(X1B)X2C,X3C,X3C      $
 X2C   X1B=X1B+360.     $
       GO TO X3C        $
 X3C   WRITE OUTPUT TAPE 5,X4C  $
 X4C   FORMAT(3H70//,4X,4HAXIS,9X,5HECCEN,6X,5HINCLN,8X,9HMEAN ANOM,3X,
       8HARG PERG,5X,7HRA NODE)$
       WRITE OUTPUT TAPE 5,X5C,X1A,X2A,X3A,X1B,X2B,X3B   $
 X5C   FORMAT(2H10,1X,F10.5,3X,F9.8,2X,4(F12.8,1X)//)$
       CALL XYZSB$
*
*PRINT OUT CHECK,CASE,LO,X,THRU X(6)$
*                 $
 PRINTR WRITE OUTPUT TAPE 5,X6C  $
 X6C   FORMAT(2H10,3X,7HL SUB 0,8X,3HAXN,9X,3HAYN,9X,2HHX,10X,2HHY,10X,
       2HHZ)   $
       WRITE OUTPUT TAPE 5,X7C,XLO,AXN,HX $
 X7C   FORMAT(2H10,6F12.8) $
```

```
C0F10   OTPT=0. $
        WRITE OUTPUT TAPE 5,X80 $
X80     FORMAT(2H10,4X,4HTIME,11X,1HX,11X,1HY,11X,1HZ,9X,5HX DOT,7X,
        5HY DOT,7X,5HZ DOT)$
        WRITE OUTPUT TAPE 5,X90,OTPT(1),(X(I),I=1,3),(XDOT(I),I=1,3)$
X90     FORMAT(2H10,7F12.8)       $
*              $
*              $
TSTRF   DO 172 I=1,5$
 172    X1STSG(I) = X1STSG(I+1) $
        IF(CODE(3))Q100,Q2,Q100$
Q2      IF(IASIZE)Q3,Q100,Q3$
Q3      REWIND  9       $
        REWIND  14      $
        WRITE OUTPUT TAPE 5,Q30$
Q30     FORMAT(2H70,44HNEW POSITIONS FOR STATIONS IN NON=ZERO DATUM/)$
        WRITE OUTPUT TAPE 5,Q31$
Q31     FORMAT(2H10,4H STA,6X,8HLATITUDE,9X,9HLONGITUDE,6X,
        8HELEV-MTS)        $
        DO(Q4)I=1,IASIZE $
        DO(Q4)J=1,3       $
        II=6+3*(I-1)+J    $
Q4      DELT(J,I)=GLSQX(II)$
        II=5     $
        DO(Q6)I=1,ISTCNT $
        STARK=STARC(II) $
        J=IAXD $
        IF(J)Q5,Q6,Q5         $
Q5      PHIRD=STARC(II-3) $
        XLAMBA=STARC(II-2)$
        OALT=STARC(II-1)  $
        SINPH=SIN(PHIRD)  $
        TEMP=(1.0-SINPH**2*EPSQD)$
        CAPC=1.0/SQRT(TEMP)     $
        CAPS=CAPC*(1.0-EPSQD)  $
        CAPZ=(CAPS+OALT)*SINPH-DELT(3,J)$
        COSPH=COS(PHIRD)        $
        XOVCT=(OALT+CAPC)*COSPH$
        COSTH=COS(XLAMBA)       $
        SINTH=SIN(XLAMBA)       $
        CAPX=XOVCT*COSTH-DELT(1,J) $
        CAPY=XOVCT*SINTH-DELT(2,J) $
        TEMP=SQRT(CAPX**2+CAPY**2)$
        TEMP=CAPZ/TEMP $
        STARC(II-3)=ATAN(TEMP*1.006814755)$
        STARC(II-2)=ATAN(CAPY/CAPX)$
        IF(CAPX(1))QQ1,QQ2,QQ2 $
QQ1     STARC(II-2)=STARC(II-2)+PI $
QQ2     CONTINUE $
        TEMP=1.0/(1.0+TEMP**2) $
        TEMP=SQRT(1.0-.0067686580*TEMP)$
        STARC(II-1)=SQRT(CAPX**2+CAPY**2+CAPZ**2)-.996609924/TEMP$
        PHIRD=STARC(II-3)/CONV  $
        XLAMBA=STARC(II-2)/CONV $
        XLAMBA=-XLAMBA $
        OALT=STARC(II-1)*XMPER  $
        WRITE OUTPUT TAPE 5,Q32,STARC(II-4),PHIRD,XLAMBA,OALT$
Q32     FORMAT(2H10,A4,2(4X,F13.8),4X,F8.2///)$
        WRITE OUTPUT TAPE 5,Q32A,CAPX(1),CAPY,CAPZ$
Q32A    FORMAT(2H10,10X,3HX= ,F11.8,6X,3HY= ,F11.8,6X,3HZ= ,F11.8)$
Q6      II=II+9       $
```

```
       DO(Q7) J=1,IOBCNT $
       READ TAPE 9,(WSTO(I),I=1,16),ASIG1,IAXD,SPHA,
       CPHA,SDEL,CCDEL,CLM,SLM,IBMUTH,EQQ $
       IF(IAXD)Q8,Q7,Q8 $
Q7     WRITE TAPE 14,(WSTO(I),I=1,16),ASIG1,IAXD,SPHA,
       CPHA,SDEL,CCDEL,CLM,SLM,IBMUTH,EQQ $
       REWIND  9 $
       REWIND  14$
       DO(Q9) J=1,IOBCNT  $
       READ  TAPE14,(WSTO(I),I=1,16),ASIG1,IAXD,SPHA,
       CPHA,SDEL,CCDEL,COSTH,SINTH,IBMUTH,EQQ$
Q9     WRITE TAPE 9,(WSTO(I),I=1,16),ASIG1,IAXD,SPHA,CPHA,SDEL,CCDEL,
       COSTH,SINTH,IBMUTH,EQQ     $
       REWIND  9 $
       GO TO 6100  $
Q8     II=1     $
       DO(Q10)K=1,ISTCNT      $
       IF(WSTO(2))E(STARC(II)),GO TO Q12 $
Q10    II=II+9 $
Q11    WRITE OUTPUT TAPE 5,Q13$
Q13    FORMAT(2H10,17HSTATION NOT FOUND)$
       STA T TAC $
       RUNOUT  $
       JMPL     5 $
       ENDTAC  $
Q12    PHIRD=STARC(II+1)   $
       XLAMBA=STARC(II+2) $
       OALT=STARC(II+3)     $
       SINPH=SIN(PHIRD)     $
       COSPH=COS(PHIRD)     $
       TEMP=1.0-SINPH**2*EPSQD $
       CAPC=1.0/SQRT(TEMP)      $
       CAPS=CAPC*(1.0-EPSQD)   $
       CAPZ=-(CAPS+OALT)*SINPH$
       XMNUTS=WSTO(4) $
       XOVCT=-(OALT+CAPC)*COSPH$
       XLMPTH=THGRC+XLAMBA      $
       THTA=XMNUTS*RPTIM+XLMPTH+EQQ $
       COSTH=COS(THTA)$
       SINTH=SIN(THTA)$
       CAPX=XOVCT*COSTH $
       CAPY=XOVCT*SINTH $
       CLM=COS(THTA-XLAMBA) $
       SLM=SIN(THTA-XLAMBA) $
       WSTO(5)=CAPX(1)   $
       WSTO(6)=CAPY   $
       WSTO(7)=CAPZ   $
       STARE=WSTO(3)$
       IF(IAXD-3)Q14,Q15,Q14 $
Q15    WSTO(8)=-CAPY*.05883447   $
       WSTO(9)=CAPX*.05883447   $
       GO TO Q7 $
Q14    IF(IAXD-2)Q16,Q7,Q16  $
Q16    IF(IAXD)Q17,Q7,Q17                  $
Q17    ASUXT=CPHA$ASUXT=SPHA     $
       XLSUZH=SDEL$DSUZT=CCDEL  $
       XLSUXH=(DSUZT*ASUYT)*(-1.0)$
       XLSUYH=ASUXT*DSUZT $
       ASUZT=0.$
       DSUYT=XLSUZH*(-ASUXT) $
       DSUXT=XLSUZH*ASUYT     $
```

```
        ESUBX=-SINT $ESUBY=COSTH$ESUBZ=0.$
        ZSUBZ=S NFR(1)
        SSUBY=- SUBZ*ESUBX $
        SSUBX=ZSUBZ*ESUBY $
        SSUBZ=-ZUSB  $
        ZSUBY=SSUBZ*ESUBX $
        ZSUBX=-SSUBZ*ESUBY$
        DO(120)  I=1,3  $
        IA=I+7$ID=I+10$IL=I+13 $
        WST(IL)=XLSUYH*ESUBX(I)+XLSUXH*SSUBX(I)+XLSUZH*ZSUBX(I)$
        WST(IA)=ASUXT*SSUBX(I)+ASUYT*ESUBX(I)+ASUZT*ZSUBX(I)$
 020    WST (ID)=DSUXT*SSUBX(I)+DSUYT*ESUBX(I)+DSUZT*ZSUBX(I)$
        GO TO 47$
 8100   EFLAG = 0.$
        IF(IRPT)8173,8173,174 $
 174    IRPT=IRPT-1$
        IF (SENSE LIGHT 1) SEC3, SEC3 $ GO DO ANOTHER CORRECTION
*START A NEW PAGE AND PRINT SIGMAS
 8173   WRITE OUTPUT TAPE 5, 1006$
 1006   FORMAT(2H10,21X,10HSIGMAS FOR,3X,9HDELTA A/A,6X,9HDELTA AXN,6X,
        9HDELTA AYN,6X,8HDELTA U0,7X,10HDELTA NODE,6X,7HDELTA I)$
        WRITE OUTPUT TAPE 5, 1007,(BDIAS(I),I=1,6)$
 1007   FORMAT(2H10,29X,6F15.9)$
*PRINT OUT SIGMAS FOR POSITIONAL ERRORS
        IF(IASIZE)P10,P50,P10  $
 P10    WRITE OUTPUT TAPE 5, 1008$
  1008  FORMAT (34H10SIGMAS   POSITION ERROR OF DATUM   )$
        L=LSTCNT  $
        J=7$
        DO 753 K = 1,IASIZE $
        BDIAS(J)=BDIAS(J)*XMPER  $
        BDIAS(J+1)=BDIAS(J+1)*XMPER  $
        BDIAS(J+2)=BDIAS(J+2)*XMPER  $
 819    WRITE OUTPUT TAPE 5,1009,IDFUF(K),BDIAS(J),BDIAS(J+1),BDIAS(J+2)
          $
 1009   FORMAT(2H10,26X,I3,4X,10HSIGMA X  =,3X,F13.2,4X,10HSIGMA Y  =,3X
        ,F13.2,4X,10HSIGMA Z  =,3X,F13.2)  $
  753   J = J + 3 $
     P50IPGCNT=IPGCNT+1  $
        REWIND 11    $
        XFLAG=0              $
  S42   READ TAPE 11,RKSTO(1),(RSUBA(I),I=1,4),(USUBA(I),I=1,4),(UX(I),I
        =1,3),( X(I),I=1,3),R,COSI,SINI,SINU,COSU,(X(I),I=1,3),
        (WX(I),I = 1,3) $
        IF(RKSTO(1)) E (HEAD), GO TO 173 $
*FORM THE 6 VECTORS FOR X,Y,Z
        DO 759 I=1,3$
  759    AVEC(I) = UX(I) * RSUBU+VX(I) * USUBU $
        XVEC(1)=AVEC(1)$YVEC(1)=AVEC(2)$ZVEC(1)=AVEC(3) $
        DO 760 I=1,3$
  760   AVEC(I) = UX(I) * RSUBA + VX(I) * USUBA $
        XVEC(2)=AVEC(1)$YVEC(2)=AVEC(2)$ZVEC(2)=AVEC(3) $
        DO 761 I=1,3$
  761     AVEC(I) = UX(I) * RXN + VX(I) * UXN     $
        XVEC(3)=AVEC(1)$YVEC(3)=AVEC(2)$ZVEC(3)=AVEC(3) $
        DO 762 I=1,3$
  762   AVEC(I) =     UX(I) * RYN + VX(I) * UYN$
        XVEC(4)=AVEC(1)$YVEC(4)=AVEC(2)$ZVEC(4)=AVEC(3) $
        DO 763 I=1,3$
  763   AVEC(I) = VX(I) * X(I)*COSI - WX(I)*X(I) *COSU*SINI $
        XVEC(5)=AVEC(1)$YVEC(5)=AVEC(2)$ZVEC(5)=AVEC(3) $
```

```
        DO 764 I=1,3$
  764    AVEC(I) =  X(1) * X(I) * SINU     $
        XVE (6)=AVE (1)$YVEC(6)=AVEC(2)$ZVEC(6)=AVEC(3) $
*TRANSP SE  ATRICES X,Y,Z
        STA T T C $
        SAVE      FMA U.HC,FMAMU.FMAM +1 $
        TMA       ADRC            $
        TM        ADR             $
        JMP       TRP.TRP         $
        TMA       ADRC            $
        TM        ADRY            $
        JMP       TRP.TRP         $
        TMA       ADRC            $
        TM        ADRZ            $
        JMP       TRP.TRP         $
*TO MULTIPLY 3  ATRICES
A       FMAMU     ADRCC@ADR1X@ADR2X$
A       FMA U     ADRCC@ADR3X@ADR4X$
A       FMAMU     ADRCC@ADR1Y@ADR2Y$
A       FMAMU     ADRCC@ADR3Y@ADR4Y$
A       FMAMU     ADRCC@ADR1Z@ADR2Z$
A       FMAMU     ADRCC@ADR3Z@ADR4Z$
        JMP       GOO   $
 ADRC   C/HLT,F C/HLT,1          $
 ADRX   C/HLT,XVEC@C/HLT,XTVEC  $
 AD Y   C/HLT,YVEC@C/HLT,YTVEC  $
 ADRZ   C/HLT,ZVEC@C/HLT,ZTVEC  $
 ADRCC  C/HLTL,1@C/HLTL,6       $
 AD 1X  C/HLTL,6@C/HLTL,XTVEC   $
 AD 1Y  C/HLTL,6@C/HLTL,YTVEC   $
 AD 1Z  C/HLTL,6@C/HLTL,ZTVEC   $
 ADR2X  C/HLTL,CVERT@C/HLTL,CXVERT $
 ADR2Y  C/HLTL,CVERT@C/HLTL,CYVERT $
 ADR2Z  C/HLTL,CVERT@C/HLTL,CZVERT $
 ADR3X  C/HLTL,1@C/HLTL,CXVERT $
 AD 3Y  C/HLTL,1@C/HLTL,CYVERT $
 AD 3Z  C/HLTL,1@C/HLTL,CZVERT $
 AD 4X  C/HLTL, VEC@C/HLTL,XCX $
 AD 4Y  C/HLTL,YVEC@C/HLTL,YCY $
 AD 4Z  C/HLTL,ZVEC@C/HLTL,ZCZ $
        EN  TAC
*FOLLOWING THE FMAMU ROUTINES
 GOON    AXSIG=S RT(XCX)*XMPER    $
         AYSIG=S RT(YCY)*XMPER    $
         AZSIG=S RT(ZCZ)*XMPER    $
         DO 765 I=1,18$
  765     CHED(I) = CHED5(I)$
         IF(XFLAG)766,767,766$
     767XFLAG=1$
         LINC=60 $
     766IF(LINC-54)769,769,768$
     768CALL HEAD(IPGCNT)$
         LINC=5$IPGCNT=IPGCNT+1 $
     769LINC=LINC+1$
*PRINT SATELLITE POSITION SIGMAS
  XX4M   WRITE OUTPUTTAPE 5,Z227,RKSTO(1),(X(I),I=1,3),AXSIG,AYSIG,AZSIG$
 1010    FORMAT(2H10,7F15.8)$
         GO TO S-2$
 173     IF(CNT. (10))175,START,175$
 175     TEMP=CNTWD(10)*(-.25068448)-THGP+360.0$
 X178    IF(TEMP)176,177,177$
```

```
176     TE  =T  P + 36 .0$
        GO TO  178  $
177     TH  =-(T  F-36 .0)$
        WRITE OUTPUT TAPE 5,XPUNH1, ,X  $
XPUNH1  FORMAT(2H70/2H10,30X,2 HRESTART INFORMATION /2H10/2H10,25X,1HD,
             14X,1HM/2H10,23X,2E16.8/2H10) $
        WRITE OUTPUT TAPE 5,XPUNH2, XLO, AXN, AYN, HX, HY, HZ $
XPUNH2  FORMAT(2H10,24X,2HLO,11X,3HAXN,11X,3HAYN,11X,2HHX,12X,2HHY,12X,
                 2HHZ/2H10,15X,6E15.6) $
        WRITE OUTPUT TAPE 5,XPUNH3,CNTWD(5)  $
XPUNH3  FORMAT(2H10/2H10,50X,9HEPOCH DAY/2H10,51X,A4)  $
         GO TO STARTS
SUBER   WRITE OUTPUT TAPE 5, SUBER2 $
SUBER2  FORMAT (2H10/1 H10SUBROUTINE ERROR  ) $
        START TAC $
        RUNOUT   $
        JMPL     2 $
RKER    TMA      C/HLT,RKBCD2 C/HLT,4$
        JMPL     6$
        RUNOUT   $
        JMPL     3$
RKBCD2  A/ERROR IN RUNGE KUTTA ROUTINES
NOSTA   TMA      C/HLT,NSBCD  C/HLT,4$
        JMPL     6$
        END TAC  $
        GOTOSTART $
        STARTTAC    $
NSBCD   A/END OF STATION DATA CARDS $
ERROR   TMA      C/HLT,OBBCD  C/HLT,4$
        JMPL     6$
        JMP      START  $
OBBCD   A/ERROR IN TAN A DIVIDING 0 BY 0$
 MXERR  TMA      C/HLT,MXBCD  C/HLT,3$
        JMPL     6$
        RUNOUT   $
         JMPL    3$
MXBCD   A/CANNOT SOLVE MATRIX$
BLAB4   TMA      C/HLT,BLAB3  C/HLT,5$
        JMPL     6$
        JMP      X130    $
BLAB3   A/STATION NUMBER            NOT FOUNDS
STARS   A/*************          $
LUNBCR  A/NO.OF UNKNOWNS EXCEEDS NO.OF OBSERVATIONS$
TSTER   TMA      C/HLT,TSTER3  C/HLT,4$
        JMPL     6$
  FINIS RUNOUT   $
        JMPL     2$
LTSTER3 A/ERROR IN TESTING TIME POINTS$
  CHED1 A/       TIME                   LATITUDE                LONGITUDE $
        A/         ALTITUDE-KM                                           $
CHED2   A/                        TIME                 X                 Y
                   Z                     X-DOT            Y-DOT          Z-DOT $
CHED3   A/CASE        SUM-        SUM2-KM/SEC      DELTA A/A      DELTA AXN
                   DELTA AYN          DELTA UO        DELTA NODE      DELTA I
                    E
CHED4   A/CASE                                            L             AXN
                         AYN                HX                HY               HZ $
CHED5   A/SIGMAS-POSITION      TIME                 X                 Y
                   Z                     SIG X(M)      SIG Y(M)        SIG Z(M)$
OCHED   A/STA                TIME - MIN        RANGE            RANGE RATE       R
                 T ASCEN          DECLIN        AZIMUTH          ALTITUDE
```

```
                   $
 OHSHD  A/SAT    STA      TIME          RHO RES-M      R A RES-M      DEC RES
                 = M    AZ RESID- M    EL RESID- M   RDOT RES-KM/S       $
 HH D    A/F D     $
*TABLE OF DAYS IN YEAR (NOTE THIS TABLE IS REVERSED)$
         PAGE      $
*LEAST SQUARES SUBROUTINE,
*
* CHGNXN SETS UP THE LEAST SQUARES ROUTINE FOR MATRIX SIZE N
*  ENTER WITH NT15 IN A
*   SETS UP THE FOLLOWING PARAMETERS  --------
* COUNTL     NT15
* COUNTR     NT39
*COUNTLR     NT15@NT39
* LSQP1      C/HLTL,TERMS+N@C/JMP,LSQMULT
* LSQP2      C/HLTL,QLSQA+N*(N+1)@  C/JMP,LSQLOD2
* LSQP3      C/HLTL,QLSQB+N@  C/HLTL,QLSQA+N
*LSQSP1      NT15
* LSQP5      (N-1)T15
* LSQSP3     NT15@NT39
* LSQDN      C/TEM,0,2@C/AIXOL,N,2     ------      PROGRAM INSTRUCTION
* LSQP6      (N+1)T15@ (N+1)T39
 CHGNXN TJM      CHGNXNX$
        TAM      COUNTL$
        SRD      24$
        TDM      COUNTR$
        AD       $
        TAM      COUNTLR$
        TMA      C/HLTL,TERMS@C/JMP,LSQMULT$
        AM       COUNTL$
        TAM      LSQP1$
        TMQ      COUNTR$
        SRQ      8$
        MM       COUNTL$
        TQA      $
        AM       COUNTL$
        AM       C/HLTL,QLSQA@C/JMP,LSQLOD2$
        TAM      LSQP2$
        TMA      C/HLTL,QLSQB@C/JMP,LSQLOC$
        AM       COUNTL$
        TAM      LSQP3$
        TMA      C/HLTR,QLSQA@C/HLTL,QLSQA$
        AM       COUNTR$
        TAM      LSQP4$
        TMA      COUNTL$
        TDM      LSQSP1$
        SM       C/HLTL,1$
        TAM      LSQP5$
        TMQ      COUNTLR$
        TDM      LSQSP3$
        TMA      C/TEM,0,2@C/AIXOL,0,2$
        AM       COUNTR$
        TAM      LSQDN$
        TMA      C/HLTL,1@C/HLTL,1$
        AM       COUNTLR$
        TAM      LSQP6$
  CHGNXNXJMP     0$
  LS     TMD     C/HLTL,P@C/HLTR,QLSQA$
         TDXPC   ,1$
         TM      F/0$
 L       RPT     600$
```

```
         TD        0,1$
LS EXECJMP         FUNCT$
LS R     TM        LS LKD$
         TD        LS  P1$
         TIJ       0$
         TJ        LSQADD$
         TMD       C/HLTR,QLSQA$
         TDXLC     ,1$
LSQLOO2TMD         LSQNP1$
         TDXLC     ,2$
         TMQ       0,2$
LSQMULTFMPR        0,2$                              BUILD A
         FAMS      0,1$
         TMD       LSQP1$
         AIXJ      1,2$
         INCAL     LSQNP1$
         INCA      LSQADD$
LLSQTMD  TMD       LSQP2$
LSQADD   AIXJ      0,1$
         TMD       C/HLTR,TERMSQC/HLTL,QLSQB$
         TDXLC     ,1$
         TDXRC     ,3$
         TMQ       0,2$
LSQLOC   FMMS      0,1$                              BUILD B
         FAMS      0,3$
         TMD       LSQP3$
         AIXJ      1,3$
         JMP       LSQEXEC$
LSQS1    TJM       LSQSX$           FILL IN A  MATRIX
         TMD       LSQP4$
         TDM       LSQP4V$
         TMD       LSQP5$
         TDM       LSQP5V$
LSQRLOOTMD         LSQP4V$
         TDXLC     ,1$
         TDXRC     ,2$
         TMD       LSQP5V$
         TDXLC     ,3$
LSQTRA   TMD       1,1$
LLSQDN   TDM       0,2$
         AIXOL     25,2 $
         TMD       C/HLT,0QC/JMP,LSQTRA$
         SIXJ      1,3$
         CAM       LSQP6$
         AMS       LSQP4V$
         CSM       C/HLTL,1$
         AMS       LSQP5V$
         JAZ       LSQSX$
         JMP       LSQRLOO$
LSQS     TJM       LSQSX$
A        FMAIN     LSQSP1  LSQSP2 $
         END TAC
*DIAGONAL ELEMENTS AND ROOTS
         IBQ=LSTCNT     $
          IF(LMODE)E(3), IBQ = 6 $
         IBQS=IBQ**2              $
         IBQ1=IBQ+1            $
         I=0$
          DO 774 K = 1, IBQS, IBQ1 $
         I=I+1$
           IF(LMODE) NE(2), GO TO 770 $
```

```
          IF (I - 6)     774, 774, 770 $
  770    BDIAG(I)=QLSQA(K)$
         BDIAS(I) = SQRT(QLSQA(K))$ SQRT(DIAG ELEM OF INVERSE MATRIX A)
   774     CONTINUE $
          IF (LMODE) E (2),     GO TO MATM $
*RE-LOCATE THE 6X6 MATRIX IN CVERT
          I = 1 $
         DO 772    = 1,6 $
         J = (M-1) * LSTCNT + 1 $       FORM CVERT,
         K = J + 5 $
         DO 771 L = J,K $
         CVERT(I) = QLSQA(L) $
  771    I = I + 1 $
  772     CONTINUE $
         START TACS
A MATM   FMAMU    LSQSP3@LSQSP4@LSQSP5 $
 LSQSX   JMP      0$
 LSQNP1  C/HLTL,TERMS@C/HLTL,0$
 LSQLWD  C/HLT,TERMS@C/HLTL,0$
 LSQP1   C/HLT,TERMS@C/JMP,LSQMULTS
 LSQSP2  C/HLTL,QLSQA@C/HLTL,LSQBUF$
 LSQSP4  C/HLTL,1@C/HLTL,QLSQA$
 LSQSP5  C/HLTL,QLSQB@C/HLTL,QLSQX$
 LSQBUF  SET      (P)+50 $
         PAGE$
         AFEND    40 $
         AFEND    40$
         NAME     BCD2BIN
 EXIT    JMP      (P)
 BCD2BIN TJM      EXIT
         TMQ      0/77
         TAM      BCDWORD
         ETA      BCDWORD
         JQO      EXIT
 TAD1    TAD
         SRAQ     2
         AD
         SRAQ     3
         EA       BCDWORD
         JQE      TAD1
         JMP      EXIT
         PAGE$
*CONVERT EXP.DEC. TO FLT. BINARY$
         AFEND    40$
         NAME     FLTFLT
 FLTFLT  TJM      FLTFLTX
         JMP      INDEX
         TJM      DCNTR $
         CM       MSIGN
         CM       CSIGN
         CM       DIGITS
         CM       CNTDIG
         CM       ADIG
         TMD      K8
         TDM      AF
         TMD      K9
         TDM      AD
         TQM      SAV
         SRAQ     18
         TQM      SAV1
         TAQ
```

```
        TIXZ    0,5
        TIXZ    0,3
        TIXZ    0,4
 AB     TIXZ    0,6
 AA     CA
        SLAQ    6
        TMD     ENE
        JAED    AC
LAF     JAZ     AC
        TMD     BLANK
        JAED    AC
        TMD     PLUS
        JAED    AC
        TMD     MINUS
 AZY    JAED    MIN
LAD     TMD     DEC
        JAED    DECA
        SM      DIC
        JAP     AC
        AM      DIC
        TMD     K7
        TDM     AF
        INCAR   CNTDIG
        TAM     DIG
        TMA     DIGITS
        SLA     2
        AM      DIGITS
        SLA     1
        AM      DIG
        TAM     DIGITS
 AC     TMD     K1
        AIXJ    1,6
        TMQ     SAV1
        TMD     K2
        AIXJ    1,5
        TMA     DCNTR $
        TMD     CNTDIG $
        JAZ     (P)+2H $
        TDM     ADIG $
        TMA     CNTDIG
        SM      ADIG
        SMS     CNTDIG
        TMQ     DIGITS
        CA
        SLAQN   12
        SLAN    35
        AM      D/47
        TAM     DIGITS
        FCAMS   DIGITS
        TQA
        AM      D/35
        FAMS    DIGITS
        TMQ     K4
        ETA     SAV
        SRD     6
        SRA     4
        AD
        SLA     1
        TMQ     K5
        EA      SAV                 EXP IN A
        TMQ     SAV
```

```
         SLA      8
         TAM      SAV              EXPT 39
         SRQ      16
         JQO      (P)+2H
         CSMS     SAV
         TMA      SAV
         AMS      CNTDIG
         JAZ      AM
         JAP      (P)+3H
         TJM      CSIGN
         CSMS     CNTDIG
         TMD      CNTDIG
         TDXRC    ,4
         TMA      F/1.
         TMQ      F/10.
AN       FMAR
         TMD      K6
         SIXJ     1,4
         TAM      SAV
         TMA      CSIGN
         JAZ      (P)+5H
         TMA      DIGITS
         FDA      SAV
         TQM      DIGITS
         JMP      (P)+3H
         TMQ      SAV
         FMMRS    DIGITS
AM       TMA      MSIGN
         JAZ      (P)+2H
         FCSMS    DIGITS
         TMA      DIGITS
         JMP      INDEXE
FLTFLTX  JMP      (P)
MIN      TJM      MSIGN
         JMP      AC
DECA     TMD      K3
         TDM      AD
         CM       DCNTR $
         JMP      AC
INDEX    TJM      INDEXX
         TXDLC    ,1
         TXDRC    ,2
         TDM      X1X
         TXDLC    ,3
         TXDRC    ,4
         TDM      X2X
         TXDLC    ,5
         TXDRC    ,6
         TDM      X3X
INDEXX   JMP      (P)
X1X       $
X2X       $
X3X       $
INDEXE   TJM      INDEXEX
         TMD      X1X
         TDXLC    ,1
         TDXRC    ,2
         TMD      X2X
         TDXLC    ,3
         TDXRC    ,4
         TMD      X3X
```

```
        TDXLC    ,5
        TDXRC    ,6
INDEXEXJMP       (P)
BLANK   0/60T47
K1      C/HLT,8@C/JMP,AA
K2      C/HLT,2@C/JMP,AB
K3      C/INCAR,ADIG@C/NOP
K4      0/7700T47
K5      0/77T47
K6      C/HLT,0@C/JMP,AN
PLUS    0/20T47
MINUS   0/40T47
K7      C/NOP,0@C/TMD,BLANK$
K8      C/JAZ,AC@C/TMD,BLANK
K9      C/TMD,DEC@C/JAED,DECA
DEC     0/33T47
ENE     W/0000000E
DIC     0/12T47
* FIXED POINT DEC. TO FLOATING BINARY$
        AFEND    40$
FXFLT   TJM      FXFLTX
        JMP      INDEX
        CM       MSIGN
        CM       DIGITS
        CM       ADIG
        TMD      X1
        TDM      BD
        TQM      SAV
        TAQ
        TIXZ     0,1
BB      TIXZ     0,2
BA      CA
        SLAQ     6
        TMD      BLANK
        JAED     BC
        TMD      PLUS
        JAED     BC
        TMD      MINUS
        JAED     BMIN
LBD     TMD      DEC
        JAED     BDEC
        TAM      DIG
        TMA      DIGITS
        SLA      2
        AM       DIGITS
        SLA      1
        AM       DIG
        TAM      DIGITS
BC      TMD      X3
        AIXJ     1,2
        TMQ      SAV
        TMQ      X4
        AIXJ     1,1
        TMQ      DIGITS
        CA
        SLAQN    12
        SLAN     35
        AM       D/47
        TAM      SAV1
        FCAMS    SAV1
        TQA
```

```
         AM       D/35
         FAMS     SAV1
         TMA      ADIG
         JAZ      BR
         TDXRC    ,1
         TMA      F/1.
         TMG      F/10.
 BN      FMAP
         TMD      X5
         SIXJ     1,1
         TAM      SAV
         TMA      SAV1
         FDAS     SAV
 BR1     TMA      MSIGN
         JAZ      (P)+2H
         FCSMS    SAV
         TMA      SAV
         JMP      INDEXE
 FXFLTX  JMP      (P)
*
 BR      TMD      SAV1
         TDM      SAV
         JMP      BR1
LX1      TMD      DEC
         JAED     BDEC
 X2      C/NOP,0@C/INCAR,ADIG
 X3      C/HLT,8@C/JMP,BA
 X4      C/HLT,2@C/JMP,BB
 X5      C/HLT,0@C/JMP,BN
 BMIN    TJM      MSIGN
         JMP      BC
 BDEC    TMD      X2
         TDM      BD
         JMP      BC
         PAGE$
         AFEND    80$
         NAME     TRP$
LRESET   HLT      $
         HLT      $
LTRP     TAD      $
         NOP      TRANS$
         TDXLC    ,1X$
         TDXRC    ,2X$
         TJM      EXIT$
         TIJ      ,1X$
         TJM      TRANS$
         TIJ      ,2X$
         TJM      TRANS+1H$
         TQM      RESET$
         ADXL     ,2X$
         TIJ      ,2X$
         TDXLC    ,1X$
         TDXRC    ,2X$
         TJM      TRP$
RTRANS   RPTAA    $
         TMD      0,1X$
         TDM      1,2X$
         INCA     RESET$
         TDXL     ,1X$
         TMD      TRP$
         AIXJ     0,1X$
```

```
 EXIT     JMP      (P)$
          ENDSUB   $
*         RUNGE KUTTA INTEGRATION ROUTINE
          NAME     RUNGE$
LEXIT     JMP      (P)$
 ERROR    JMP      (P)$
 RUNGE    TJM      ERROR$               SET ERROR EXIT
          TXDLC    ,3X$                 SAVE 3X THRU 0X
          TXDRC    ,4X$
          TDM      X3X4$
          TXDLC    ,5X$
          TXDRC    ,6X$
          TDM      X5X6$
          TXDLC    ,7X$
          TXDRC    ,0X$
          TDM      X7X0$
*             SET UP INPUT PARAMETERS
LXSET     TAD      $                    GET A PARAMETER
          TDXRC    ,1X$
          TDXLC    ,2X$
          TXDRC    ,1X$
          TJM      CTL$                 SET CONTROL ENTRY
          TXDLC    ,2X$
          TJM      DER1$                SET DERIV ENTRIES
          TJM      DER2$
          TJM      DER3$
          TJM      DER4$
          TJM      DER5$
          TQD      $                    GET Q PARAMETER
          TDXLC    ,1X$
          TXDLC    ,1X$
          TJM      TESTM$           SET  MODE ADDRESS
          TCXSC    ,1X$
          TXDL     ,1X$
          TJM      CX$                 X FL
          TJM      EX$
          TCXZ     ,1X$
          TXDL     ,1X$
          TJM      BX$                  X FX
          TJM      IX$
          AIXCR    1,1X$
          TXDL     ,1X$
          TJM      ADXP$                PREV DELTA-X FX
          TJM      BDXP$
          AIXCR    1,1X$
          TXDL     ,1X$
          TJM      AXZ$                 X-ZERO FX
          TJM      IXZ$
          TJM      BXZ$
          TCXSC    ,1X$
          TXDL     ,1X$
          TJM      HDX1$                DELTA-X FL
          TJM      HDX2$
          TJM      HDX3$
          TCXZ     ,1X$
          TXDL     ,1X$
          TJM      ADX$                 DELTA-X FX
          TJM      BDX$
          AIXCR    1,1X$
          TXDL     ,1X$
          TJM      XSET$                Y FX
```

```
TQD      $
ADXR     ,1X$
TXDL     ,1X$       Y+N TO JA
TMD      FILL$      SET LOOP TESTERS
TDXRC    ,1X$
RPTN     9$
TJM      ,1X$
TQD      $
TDXR     ,1X$       GET N
TCXS     ,1X$
TXDL     ,1X$
TJM      DN$        N FL
TJM      EN$
TJM      FN$
TCXZ     ,1X$
TXDL     ,1X$
TJM      ALOOP$     N FX
TJM      CN$
TQD      $
ADXR     ,1X$
TXDL     ,1X$
TJM      A2N$       2N FX
TJM      B2N$
TJM      I2N$
TQD      $
ADXR     ,1X$
TXDL     ,1X$
TJM      B3N$       3N FX
TJM      D3N$
TJM      E3N$
TJM      GLOOP$
TJM      I3N$
TCXS     ,1X$
TXDL     ,1X$
TJM      CLOOP$     3N FL
TJM      DLOOP$
TJM      ELOOP$
TJM      F3N$
TJM      H3N$
TQD      $
ADXR     ,1X$
TXDL     ,1X$
TJM      H4N$       4N FL
TCXZ     ,1X$
TXDL     ,1X$
TJM      A4N$       4N FX
TJM      G4N$
TQD      $
ADXR     ,1X$
TXDL     ,1X$
TJM      A5N$       5N FX
TJM      B5N$
TJM      ILOOP$
TCXS     ,1X$
TXDL     ,1X$
TJM      H5N$       5N FL
TQD      $
ADXR     ,1X$
TXDL     ,1X$
TJM      D6N$       6N FL
TJM      E6N$
```

```
         TJM      F6N$
         TCXZ     ,1X$
         TXDL     ,1X$
         TJM      C6N$               6N FX
         TQD      $
         ADXR     ,1X$
         TCXS     ,1X$
         TXDL     ,1X$
         TJM      H7N$               7N FL
         TQD      $
         ADXR     ,1X$
         TCXZ     ,1X$
         TXDL     ,1X$
         TJM      HLOOP$             8N FX
         TQD      $
         ADXR     ,1X$
         TXDL     ,1X$
         TJM      B9N$               9N FX
         TJM      I9N$
         TCXS     ,1X$
         TXDL     ,1X$
         TJM      F9N$               9N FL
         TJM      H9N$
         TMD      JUMP$              SET NO DECREASE
         JQE      (P)+1$             TEST FOR DECREASE LOOP
         TMD      RETURN$            SET DECREASE
         TDXRC    ,1X$
         TXDRC    ,1X$
         TJM      OPTJ$              SET OPTIONAL JUMP
         TMA      EXIT$
         AM       FBIT$
         TAD      $
         TDXRC    ,1X$
         TXDRC    ,1X$
         TJM      EXIT$              SET NORMAL EXIT
* ACTUAL RUNGE-KUTTA INTEGRATION STARTS HERE
 RESTARTTMD       DONE$              SET SWITCH PLUS
         TDM      SWITCH$
         TMD      XSET$              SET REGIONS TO ZERO
         TDXLC    ,1X$               FOR INITIAL PRINT
         TMQ      F/0$               GET F.P. ZERO
 ALOOP   TQM      ,1X$         CLEAR Y-HALF
 A2N     TDM      ,1X$               Y-ZERO
 A4N     TDM      ,1X$               D-HALF
 A5N     TDM      ,1X$               D-ZERO
         TMD      TEST0$
         AIXJ     1,1X$              LOOP TIL DONE
 AXZ     TQM      $            CLEAR X-ZERO
 ADX     TMD      $             MOVE DELTA-X TO
 ADXP    TDM      $                  PREVIOUS DELTA-X
 DER1    JMP      (P)$               ENTER DERIV
LCTINU   TMD      CTL$               SET INDEX FOR
         TDXR     ,7X$               VARIABLE RETURN
 CTL     JMP      (P)$               ENTER CONTROL
         JMP      RETURN$            DUMMY ADDRESS
 BX      TMD      $             MOVE X TO X-ZERO
 BXZ     TDM      $
         TMD      XSET$              SET LOOP
         TDXLC    ,1X$
         TMQ      F/0$               GET F.P. ZERO
 BLOOP   TMD      0,1X$         MOVE Y TO Y-ZERO
```

```
B2N      TDM      ,1X$
B3N      TMD      ,1X$                D TO D-ZERO
B5N      TDM      ,1X$
B9N      TQM      ,1X$         CLEAR  DELTA-Y ACCUMULATOR
         TMD      TEST1$
         AIXJ     1,1X$               LOOP TIL DONE
BDX      TMA      $                   IF DELTA-X = 0
         JAFC     XERROR$             MAKE ERROR EXIT
BDXP     TAM      $             MOVE  DELTA-X TO PREVIOUS DX
         FDA      F/2$                FORM DX/2
         TQM      HDELT$
         TDA      $
         FDA      F/2$                FORM DX/4
         TQM      QDELT$
STEP1    TMD      XSET$               SET LOOP FOR STEP 1
         TDXLC    ,1X$
         TMQ      QDELT$              GET DX/4
CLOOP    FMMR     ,1X$                D*DX/4
C6N      TAM      ,1X$                TO CUMULATIVE
         TMD      0,1X$         MOVE  Y TO Y-HALF
CN       TDM      ,1X$
         FAMS     0,1X$               Y+D*DX/4 TO Y
         TMD      TEST2$
         AIXJ     1,1X$               LOOP TIL DONE
         TQA      $
CX       FAMS     $                   X+DX/4 TO X
DER2     JMP      (P)$                ENTER DERIV
         TMD      XSET$               SET LOOP FOR STEP 2
         TDXLC    ,1X$
         TMQ      QDELT$              GET DX/4
DLOOP    FMMRS    ,1X$                D*DX/4 TO D
DN       FAM      ,1X$                D*DX/4+Y-HALF TO Y
         TAM      0,1X$
D3N      TMA      ,1X$
         FAD      $                   D*DX/2+CUMULATIVE
D6N      FAMS     ,1X$                TO CUMULATIVE
         TMD      TEST3$
         AIXJ     1,1X$               LOOP TIL DONE
DER3     JMP      (P)$                ENTER DERIV
         TMQ      HDELT$              GET DX/2
         TMD      XSET$               SET LOOP FOR STEP 3
         TDXLC    ,1X$
ELOOP    FMMRS    ,1X$                D*DX/2 TO D
EN       FAM      ,1X$                D*DX/2+Y-HALF TO Y
         TAM      0,1X$
E3N      TMA      ,1X$                D*DX/2+CUMULATIVE
E6N      FAMS     ,1X$                TO CUMULATIVE
         TMD      TEST4$
         AIXJ     1,1X$               LOOP TIL DONE
         TMA      QDELT$              X + DX/4 TO X
EX       FAMS     $
DER4     JMP      (P)$                ENTER DERIV
         TMD      XSET$               SET LOOP FOR STEP 4
         TDXLC    ,1X$
FLOOP    TMQ      QDELT$              GET DX/4
F3N      FMM      ,1X$                (D*DX/4 + CUMUL.)/3
F6N      FAM      ,1X$                + Y-HALF TO Y
         FDA      F/3$
         TQA      $
FN       FAM      ,1X$
         TAM      0,1X$
```

```
        TQA     $                 ADD HALF-STEP INCREMENT
F9N     FAMS    ,1X$              TO DELTA-Y ACCUMULATOR
        TMD     TEST5$
        AIXJ    1,1X$             LOOP TIL DONE
DER5    JMP     (P)$              ENTER DERIV
        CSMS    SWITCH$           FLIP SWITCH
        JAP     TESTM$            IF FULL STEP, TO MODE TEST
        TMD     XSET$             SET LOOP FOR D MOVE
        TDXLC   ,1X$
GLOOP   TMD     ,1X$         MOVE D TO D-HALF
G4N     TDM     ,1X$
        TMD     TEST6$
        AIXJ    1,1X$             LOOP TIL DONE
        JMP     STEP1$            TO 2ND HALF-STEP
TESTM   TMA     $                 GET MODE
        JAZ     (P)+1$            VARIABLE-TO FIND MAX
        JMP     CTINU$            FIXED-TO CONTROL ENTRY
        TMD     F/0$
        TDM     MAX$              F.P. ZERO TO MAX
        TMD     XSET$             SET MAX ERROR LOOP
        TDXLC   ,1X$
HLOOP   TMQ     ,1X$              GET REL ERR CONTROL
        FMMA    0,1X$             R*Y-ABS
H7N     FAM     ,1X$              ADD ABS ERR CONTROL
        TAM     QDELT$            SAVE ERR CONTROL TERM
        TMD     F/0$              IF ZERO
        JAED    DERROR$           MAKE ERROR EXIT
        TMQ     F/4$              FORM SIMPSONS RULE DY
H4N     FMM     ,1X$              (4*D-HALF
H3N     FAM     ,1X$              +D
H5N     FAM     ,1X$              +D-ZERO)/3
        FDA     F/3$
        FMM     HDELT$            *DX/2
H9N     FSM     ,1X$              SUBTRACT RK DELTA-Y
        FDA     QDELT$            FORM ERROR TERM
        FCAQA   $                 ABS VALUE TO A
        TMQ     MAX$              TEST MAX
        JAGQF   (P)+1$            TO SET NEW MAX
        JMP     (P)+1$            TO LOOP TEST
        TAM     MAX$              STORE NEW MAX
        TMD     TEST7$
        AIXJ    1,1X$             LOOP TIL DONE
        TMA     MAX$              GET MAX ERROR TERM
        TMQ     F/1$              TEST 1.0
        JAED    DC$               TO DECREASE AND CONTINUE
        JAGQF   DB$               TO DECR. AND BACKUP
        TMQ     F/.75$            TEST 0.75
        JAED    CTINU$            TO CONT.
        JAGQF   DC$               TO DECR. AND CONT.
        TMQ     F/.075$           TEST 0.075
        JAGQF   CTINU$            TO CONT.
        TMQ     RT10$             INCREASE AND CONT.
HDX1    FMMS    $
        JMP     CTINU$
DB      TMQ     RRT10$            DECREASE AND BACKUP
HDX2    FMMS    $
OPTJ    JMP     (P)$              OPTIONAL DECREASE LOOP
        TMQ     F/.1$
        FMMS    MAX$              MAX/10 TO MAX
        TMQ     F/1$
        JAGQF   DB$               TO DECREASE AGAIN
```

```
         JMP      BAKUP$              OK
 DC      TMG      RRT10$              DECREASE AND CONT.
 HMX3    FMVS     $
         JMP      CTINU$
 BAKUP   TMD      XSET$               SET LOOP
         TDXLC    ,1X$
         TMQ      F/0$                GET F.P. ZERO
 ILOOP   TMD      ,1X$           MOVE D-ZERO TO D
 I3M     TDM      ,1X$
 I2M     TMD      ,1X$                Y-ZERO TO Y
         TDM      0,1X$
 I9M     TQM      ,1X$                CLEAR DELTA-Y
         TMD      TEST8$
         AIXJ     1,1X$               LOOP TIL DONE
 IXZ     TMD      $              MOVE X-ZERO TO X
 IX      TDM      $
         JMP      BDX$                TO REPEAT STEP
RFILL    HLTR     TEST0$              ADDRESS DUMMY
LRETURN  JMP      BX$                 NORMAL CONTINUE
         JMP      OPTJ+1H$
LJUMP    JMP      BAKUP$
         JMP      BAKUP$
LFBIT    JMP      RESTART$
         HLTP     0$
LDONE    JMP      RESET$
         JMP      EXIT$
 XERROR  CSM      DONE$               SET A-SIGN MINUS
         JMP      RESET$              FOR DELTA-X=0
         JMP      ERROR$              TO ERROR EXIT
 DERROR  CAM      DONE$               SET A-SIGN PLUS
         JMP      XERROR+1H$          ERR CONTROL TERM=0
 RESET   TJM      BACK$               RESTORE X-REGS
         TMD      X3X4$
         TDXLC    ,3X$
         TDXRC    ,4X$
         TMD      X5X6$
         TDXLC    ,5X$
         TDXRC    ,6X$
         TMD      X7X0$
         TDXLC    ,7X$
         TDXRC    ,0X$
 BACK    JMP      (P)$
 TEST0   HLT      0$                  LOOP CONTROLS
         JMP      ALOOP$
 TEST1   HLT      0$
         JMP      BLOOP$
 TEST2   HLT      0$
         JMP      CLOOP$
 TEST3   HLT      0$
         JMP      DLOOP$
 TEST4   HLT      0$
         JMP      ELOOP$
 TEST5   HLT      0$
         JMP      FLOOP$
 TEST6   HLT      0$
         JMP      GLOOP$
 TEST7   HLT      0$
         JMP      HLOOP$
 TEST8   HLT      0$
         JMP      ILOOP$
 RT10    F/1.584893193$
```

```
RRT10  F/.6309573443$
       NAME    0001PRO $
       END TAC $
       END     $
```

XIII

REFERENCES

1. Research In Geodesy and Gravity, Computer Programs for Orbit Correction and Station Location; AFCRL 62-892, C.G. Hilton, J.E. Evans, L. Nicola (1962)

2. Handbook of Astronautical Engineering; Koelle, H.H. (1961); McGraw-Hill.

3. Optical Generator Program; AFCRL-63-445, H.R. Kahler, R.M. Moroney, W.T. Nixon (1963)

4. An Introduction to Astrodynamics; Academic Press; Baker and Makemson (1960)

5. The American Ephemeris and Nautical Almanac; United States Government Printing Office, Washington, D.C.

AF Cambridge Research Laboratories, Bedford, Mass.
Report No. AFCRL-64-55
DIFFERENTIAL ORBIT CORRECTION AND STATION LOCATOR PROGRAM — "DOC II"
Scientific Report No. 3, February 1964
230 p., incl., analysis, flow charts, program listing and operating instructions. Unclassified Report

The analysis and programming associated with performing a differential correction to orbital elements and/or station positions from weighted observations of range, range rate, azimuth-elevation and/or right ascension-declination on close earth satellites are described. The ephemeris computation considers perturbations due to the earth's gravitational potential as described by a spherical harmonic representation through n = m = 5, the lunar gravitational potential, air drag and radiation pressure. All observations may be simulated in punched card form for the purposes of conducting geometric studies.

UNCLASSIFIED

1. Astronomy & Geophysics
2. Mathematics
3. Celestial Mechanics

I. AF Project
II. Contract AF19(628)2379
III. Wolf Research and Development Corp.
IV. Hussey, I. M.
Moroney, R. M.
V. In DDC collection

UNCLASSIFIED

AF Cambridge Research Laboratories, Bedford, Mass.
Report No. AFCRL-64-55
DIFFERENTIAL ORBIT CORRECTION AND STATION LOCATOR PROGRAM — "DOC II"
Scientific Report No. 3, February 1964
230 p., incl., analysis, flow charts, program listing and operating instructions. Unclassified Report

The analysis and programming associated with performing a differential correction to orbital elements and/or station positions from weighted observations of range, range rate, azimuth-elevation and/or right ascension-declination on close earth satellites are described. The ephemeris computation considers perturbations due to the earth's gravitational potential as described by a spherical harmonic representation through n = m = 5, the lunar gravitational potential, air drag and radiation pressure. All observations may be simulated in punched card form for the purpose of conducting geometric studies.

UNCLASSIFIED

1. Astronomy & Geophysics
2. Mathematics
3. Celestial Mechanics

I. AF Project
II. Contract AF19(628)2379
III. Wolf Research and Development Corp.
IV. Hussey, I. M.
Moroney, R. M.
V. In DDC collection

UNCLASSIFIED

AF Cambridge Research Laboratories, Bedford, Mass.
Report No. AFCRL-64-55
DIFFERENTIAL ORBIT CORRECTION AND STATION LOCATOR PROGRAM — "DOC II"
Scientific Report No. 3, February 1964
230 p., incl., analysis, flow charts, program listing and operating instructions. Unclassified Report

The analysis and programming associated with performing a differential correction to orbital elements and/or station positions from weighted observations of range, range rate, azimuth-elevation and/or right ascension-declination on close earth satellites are described. The ephemeris computation considers perturbations due to the earth's gravitational potential as described by a spherical harmonic representation through n = m = 5, the lunar gravitational potential, air drag and radiation pressure. All observations may be simulated in punched card form for the purposes of conducting geometric studies.

UNCLASSIFIED

1. Astronomy & Geophysics
2. Mathematics
3. Celestial Mechanics

I. AF Project
II. Contract AF19(628)2379
III. Wolf Research and Development Corp.
IV. Hussey, I. M.
Moroney, R. M.
V. In DDC collection

UNCLASSIFIED

AF Cambridge Research Laboratories, Bedford, Mass.
Report No. AFCRL-64-55
DIFFERENTIAL ORBIT CORRECTION AND STATION LOCATOR PROGRAM — "DOC II"
Scientific Report No. 3, February 1964
230 p., incl., analysis, flow charts, program listing and operating instructions. Unclassified Report

The analysis and programming associated with performing a differential correction to orbital elements and/or station positions from weighted observations of range, range rate, azimuth-elevation and/or right ascension-declination on close earth satellites are described. The ephemeris computation considers perturbations due to the earth's gravitational potential as described by a spherical harmonic representation through n = m = 5, the lunar gravitational potential, air drag and radiation pressure. All observations may be simulated in punched card form for the purposes of conducting geometric studies.

UNCLASSIFIED

1. Astronomy & Geophysics
2. Mathematics
3. Celestial Mechanics

I. AF Project
II. Contract AF19(628)2379
III. Wolf Research and Development Corp.
IV. Hussey, I. M.
Moroney, R. M.
V. In DDC collection

UNCLASSIFIED

UNCLASSIFIED

UNCLASSIFIED